HLT-NAACL-06

Analyzing Conversations in Text and Speech (ACTS)

Proceedings of the Workshop

08 June 2006
New York City, New York, U.S.A.

Production and Manufacturing by
Omnipress Inc.
2600 Anderson Street
Madison, WI 53704

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
209 N. Eighth Street
Stroudsburg, PA 18360
USA
Tel: +1-570-476-8006
Fax: +1-570-476-0860
acl@aclweb.org

Introduction

Welcome to the HLT-NAACL Workshop on Analyzing Conversations in Text and Speech (ACTS). We received 21 submissions, and due to a rigerous review process, we rejected 11.

Organizers:

Eduard Hovy, USC/Information Sciences Institute
Klaus Zechner, Educational Testing Services
Liang Zhou, USC/Information Sciences Institute

Program Committee:

Regina Barzilay, Massachusetts Institute of Technology
Bob Frederking, Carnegie Mellon University
Graeme Hirst, University of Toronto
Anton Leuski, USC/Institute for Creative Technologies
Chin-Yew Lin, Microsoft Research Asia
Ani Nenkova, Stanford University
Doug Oard, University of Maryland
Mari Ostendorf, University of Washington
Gerald Penn, University of Toronto
Dragomir Radev, University of Michigan
Alex Rudnicky, Carnegie Mellon University
David Traum, USC/Institute for Creative Technologies
Ming Zhou, Microsoft Research Asia

Table of Contents

Conference Program

Wednesday, June 29, 2005

8:45–9:00 Opening Remarks

9:00–9:30 *Prosodic Correlates of Rhetorical Relations*
Gabriel Murray, Maite Taboada and Steve Renals

9:30–10:00 *Off-Topic Detection in Conversational Telephone Speech*
Robin Stewart, Andrea Danyluk and Yang Liu

10:00–10:30 *Computational Measures for Language Similarity Across Time in Online Communities*
David Huffaker, Joseph Jorgensen, Francisco Iacobelli, Paul Tepper and Justine Cassell

10:30–11:00 Break

11:00–11:30 *You Are What You Say: Using Meeting Participants' Speech to Detect their Roles and Expertise*
Satanjeev Banerjee and Alexander Rudnicky

11:30–11:50 *Shallow Discourse Structure for Action Item Detection*
Matthew Purver, Patrick Ehlen and John Niekrasz

11:50–12:30 Short Presentations and Discussions.

12:30–2:00 Lunch

2:00–2:30 *Improving "Email Speech Acts" Analysis via N-gram Selection*
Vitor Carvalho and William Cohen

2:30–3:00 *Topic-Segmentation of Dialogue*
Jaime Arguello and Carolyn Rose

3:00–3:30 *ChAT: A Time-Linked System for Conversational Analysis*
michelle gregory, doug love, stuart rose and anne schur

3:30–4:00 Break

Wednesday, June 29, 2005 (continued)

4:00–4:20 *Pragmatic Discourse Representation Theory*
Yafa Al-Raheb

4:20–4:50 *Retrospective Analysis of Communication Events - Understanding the Dynamics of Collaborative Multi-Party Discourse*
Andrew Cowell, Jereme Haack and Adrienne Andrew

4:50–5:00 Closing Remarks

Prosodic Correlates of Rhetorical Relations

Gabriel Murray
Centre for Speech Technology Research
University of Edinburgh
Edinburgh EH8 9LW
gabriel.murray@ed.ac.uk

Maite Taboada
Dept. of Linguistics
Simon Fraser University
Vancouver V5A 1S6
mtaboada@sfu.ca

Steve Renals
Centre for Speech Technology Research
University of Edinburgh
Edinburgh EH8 9LW
s.renals@ed.ac.uk

Abstract

This paper investigates the usefulness of prosodic features in classifying rhetorical relations between utterances in meeting recordings. Five rhetorical relations of *contrast*, *elaboration*, *summary*, *question* and *cause* are explored. Three training methods - supervised, unsupervised, and combined - are compared, and classification is carried out using support vector machines. The results of this pilot study are encouraging but mixed, with pairwise classification achieving an average of 68% accuracy in discerning between relation pairs using only prosodic features, but multi-class classification performing only slightly better than chance.

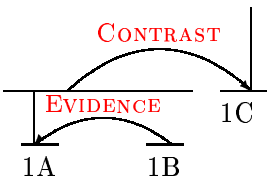
1 Introduction

Rhetorical Structure Theory (RST) (Mann and Thompson, 1988) attempts to describe a given text in terms of its coherence, i.e. how it is that the parts of the text are related to one another and how each part plays a role. Two adjacent text spans will often exhibit a nucleus-satellite relationship, where the satellite plays a role that is relative to the nucleus. For example, one sentence might make a claim and the following sentence give evidence for the claim, with the second sentence being a satellite and the *evidence* relation existing between the two spans. In a text containing many sentences, these nucleus-satellite pairs can be built up to produce a document-wide rhetorical tree. Figure 1 gives an example of a rhetorical tree for a three-sentence text[1].

Theories such as RST have been popular for some time as a way of describing the multi-levelled rhetorical relations that exist in text, with relevant applications such as automatic summarization (Marcu, 1997) and natural language generation (Knott and Dale, 1996). However, implementing automatic rhetorical parsers has been a problematic area of research. Techniques that rely heavily on explicit signals, such as discourse markers, are of limited use both because only a small percentage of rhetorical relations are signalled explicitly and because explicit markers can be ambiguous. RST trees are binary branching trees distinguishing between nuclei and satellites, and automatically determining nuclearity is also far from trivial. Furthermore, there are some documents which are simply not amenable to being described by a document-wide rhetorical tree (Mann and Thompson, 1988). Finally, sometimes more than one relation can hold between two given units (Moore and Pollack, 1992). Given the problems of automatically parsing text for rhetorical relations, it seems prohibitively difficult to attempt rhetorical parsing of speech documents - data which are marked by disfluencies, low information density, and sometimes little cohesion. For that reason, this pilot study sets out a comparatively modest task: to determine whether one of five relations holds between two adjacent dialogue acts in meeting speech. All relations are of the form nucleus-satellite, and the five relation types are *contrast*,

[1] *Contrast* is in fact often realized with a multi-nuclear structure

Proceedings of the Analyzing Conversations in Text and Speech (ACTS) Workshop at HLT-NAACL 2006, pages 1–7,
New York City, New York, June 2006.



[I love coffee.]^{1A} [I drink it every morning.]^{1B}

[But my brother has never even tried it.]^{1C}

Figure 1: *Sample RST tree*

elaboration, *cause*, *question* and *summary*. This work solely investigates the usefulness of prosodic features in classifying these five relations, rather than relying on discourse or lexical cues. A central motivation for this study is the hope that rhetorical parsing using prosodic features might aid an automatic summarization system.

2 Previous Research

Early work on automatic RST analysis relied heavily on discourse cues to identify relations (Corston-Oliver, 1998; Knott and Sanders, 1998; Marcu, 1997; Marcu, 1999; Marcu, 2000) (e.g., "however" signaling an *antithesis* or *contrast* relation. As mentioned above, this approach is limited by the fact that rhetorical relations are often not explicitly signalled, and discourse markers can nevertheless be ambiguous. A novel approach was described in (Marcu and Echihabi, 2002), which used an unsupervised training technique, extracting relations that were explicitly and unamibiguously signalled and automatically labelling those examples as the training set. This unsupervised technique allowed the authors to label a very large amount of data and pairs of words found in the nucleus and satellite as the features of interest. The authors reported very encouraging pairwise classification results using these word-pair features, though subsequent work using the same bootstrapping technique has fared less well (Sporleder and Lascarides, to appear 2006).

There is little precedent for applying RST to speech dialogues, though (Taboada, 2004) describes rhetorical analyses of Spanish and English spoken dialogues, with in-depth corpus analyses of discourse markers and their corresponding relations. The work in (Noordman et al., 1999) uses short read texts to explore the relationship between prosody and the level of hierarchy in an RST tree. The authors report that higher levels in the hierarchy are associated with longer pause durations and higher pitch. Similar results are reported in (den Ouden, 2004), who additionally found significant prosodic differences between causal and non-causal relations and between semantic and pragmatic relations.

Litman and Hirschberg (1990) investigated whether prosodic features could be used to disambiguate *sentential* versus *discourse* instances of certain discourse markers such as "incidentally." Passonneau and Litman (1997) explored the discourse structure of spontaneous narrative monologues, with a particular interest in both manual and automatic segmentation of narratives into coherent discourse units, using both lexical and prosodic features. Grosz and Hirschberg (1992) found that read AP news stories annotated for discourse structure in the Grosz and Sidner (1986) framework showed strong correlations between prosodic features and both global and local structure. Also in the Grosz and Sidner framework, Hirschberg and Nakatani (1996) found that utterances from direction-giving monologues significantly differed in prosody depending on whether they appeared as segment-intial, segment-medial or segment-final.

3 Defining the Relations

Following Marcu and Echihabi's work, we included *contrast*, *elaboration* and *cause* relations in our research. We chose to exclude *condition* because it is always explicitly signalled and therefore trivial for classification purposes. We also include a *summary* relation, which is of particular interest here because it is hoped that classification of rhetorical relations will aid an automatic speech summarization system. As in Segmented Discourse Representation Theory (SDRT) (Asher and Lascarides, 2004), an alternative framework for representing text structure, we included *question/answer* to our relations list. All training and testing pairs consist of a nucleus followed by a satellite, and the relations are defined as follows:

- **Contrast:** The information in the satellite contradicts or is an exception to the information in the nucleus. Example:
  - Speaker 1: *You use it as a tool*
    Speaker 1: *Not an end user*
- **Elaboration:** The information from the nucleus is discussed in greater detail in the satellite. Example:
  - Speaker 1: *The last time I looked at it was a while ago*
    Speaker 1: *Probably a year ago*
- **Cause:** The situation described in the satellite results from the situation described in the nucleus. Example:
  - Speaker 1: *So the GPS has crashed as well*
    Speaker 1: *So the first person has to ask you where you are*
- **Summary:** The information in the satellite is semantically equivalent to the information in the nucleus. It is not necessarily more succinct. Example:
  - Speaker 1: *The whole point is that the text and lattice are isomorphic*
    Speaker 1: *They represent each other completely*
- **Question/Answer:** The satellite fulfills an information need explicitly stated in the nucleus. Example:
  - Speaker 1: *What does the P stand for anyway?*
    Speaker 2: *I have no idea*

We also took the simplifying step of concentrating only on dialogue acts which did not internally contain such relations as defined above, which could confound the analysis. For example, a dialogue act might serve as a *contrast* to the preceding dialogue act while also containing a *cause* relation within its own text span.

4 Experimental Setup

4.1 Corpus Description

All data was taken from the ICSI Meetings corpus (Janin et al., 2003), a corpus of 75 unrestricted domain meetings averaging about an hour in length each. Both native and non-native English speakers participate in the meetings. The following experiments used manual meeting transcripts and relied on manual dialogue act segmentation (Shriberg et al., 2004). A given meeting can contain between 1000 and 1600 dialogue acts. All rhetorical relation examples in the training and test sets are pairs of adjacent dialogue acts.

4.2 Features

Seventy-five prosodic features were extracted in all, relating to pitch (or *F0*) contour, pitch variance, energy, rate-of-speech, pause and duration. To approximate the pitch contour of a dialogue act, we measure the pitch slope at multiple points within the dialogue act, e.g., the overall slope, the slope of the first 100 and 200 ms, last 100 and 200 ms, first half and second half of the dialogue act, and each quarter of the dialogue act. The pitch standard deviation is measured at the same dialogue act subsections. For each of the four quarters of the dialogue act, the energy level is measured and compared to the overall dialogue act energy level, and the number of silent frames are totalled for each quarter of the dialogue act as well. The maximum F0 for each dialogue act is included, as are the length of the dialogue act both in seconds and in number of words. A very rough rate-of-speech feature is employed, consisting of the number of words divided by the length of the dialogue act in seconds. We also include a feature of pause length between the nucleus and the satellite, as well as a feature indicating whether or not the speakers of the nucleus and the satellite are the same. Finally, the cosine similarity of the nucleus feature vector and satellite feature vector is included, which constitutes a measurement of the general prosodic similarity between the two dialogue acts. The motivation for this last feature is that some relations such as *question* would be expected to have very different prosodic characteristics in the satellite versus the nucleus, whereas other relations such as *summary* might have a nucleus and satellite with very similar

prosody to each other.

While there are certainly informative lexical cues to be exploited based on previous research, this pilot study is expressly interested in how efficient prosody alone is in automatically classifying such rhetorical relations. For that reason, the feature set is limited solely to the prosodic characteristics described above.

4.3 Training Data

Using the PyML machine learning tool[2], support vector machines with polynomial kernels were trained on multiple training sets described below, using the default libsvm solver[3], a sequential minimal optimization (SMO) method. Feature normalization and feature subset selection using recursive feature elimination were carried out on the data. The following subsections describe the various training approaches we experimented with.

4.3.1 Manually Annotated Data

For the first experiment, a very small set of manually labelled relations was constructed. Forty examples of each relation were annotated, for a total training set of 200 examples. Each relation has training examples that are explicitly and non-explicitly signalled, since we want to discover prosodic cues for each relation that are not dependent on how lexically explicit the relation tends to be. The percentage of either unsignalled or amibiguously signalled relations across all of the manually-labelled datasets is about 57%, though this varies very much depending on the relation. For example, only just over 20% of *questions* are unsignalled or ambiguously signalled whereas nearly 70% of *elaborations* are unsignalled.

4.3.2 Unsupervised

Following Marcu and Echihabi, we employ a bootstrapping technique wherein we extract cases which are explicitly signalled lexically and use those as our automatically labelled training set. Because those lexical cues are sometimes ambiguous or misleading, the data will necessarily be noisy, but this approach allows us to create a large training set without the time and cost of manual annotation. Whereas Marcu and Echihabi used these templates to extract relation examples and learn further lexical information about the relation pairs, we are using similar templates based on discourse markers but subsequently exploring the extracted relation pairs in terms of prosodic features. Three hundred examples of each relation were extracted and automatically labelled, for a training set of 1500 examples, more than ten times the size of the manually labelled training set. Examples of the explicit lexical cues used to construct the training set are provided in Table 1:

Relation	Nucleus	Satellite
Contrast	...	*However...*
	...	*But...*
	...	*Except...*
	...	*Although...*
Cause	...	*Therefore...*
	...	*As a result...*
	...	*And so...*
	...	*Subsequently...*
Elaboration	...	*Which...*
	...	*For Example...*
	...	*Specifically...*
Summary	...	*Basically...*
	...	*In other words...*
	...	*I mean...*
	...	*In short...*
Q/A	*Why/What/Where/When Who/Did/Is/Are*	...

Table 1: Templates for Unsupervised Method

4.3.3 Combined

Finally, the two training sets discussed above were combined to create a set of 1700 training examples.

4.4 Development and Testing Data

For the development set, 35 examples of each relation were annotated, for a total set size of 175 examples. We repeatedly tested on the development set as we increased the prosodic database and experimented with various classifier types. The smaller final test set consists of 15 examples of each relation, for a total set size of 75 examples. Both the test set and development set consist of explicitly and non-explicitly signalled relations. As mentioned above, the percentage of either unsignalled or amibiguously signalled relations across all of the manually-labelled datasets is about 57%

Both pairwise and multi-class classification were

[2]http://pyml.sourceforge.net

[3]http://www.csie.ntu.edu.tw/ cjlin/libsvm/

Relation Pair	Super.	Unsuper.	Combo
Contrast/Cause	0.60	0.67	0.64
Contrast/Summary	0.63	0.57	0.60
Contrast/Question	0.74	0.73	0.80
Contrast/Elaboration	0.61	0.53	0.56
Cause/Summary	0.59	0.60	0.69
Cause/Question	0.84	0.77	0.81
Cause/Elaboration	0.59	0.54	0.56
Summary/Question	0.59	0.60	0.63
Summary/Elaboration	0.70	0.63	0.70
Elaboration/Question	0.90	0.73	0.84
AVERAGE:	**68%**	**64%**	**68%**

Table 2: Pairwise Results on Development Set

carried out. The former set of experiments simply aimed to determine which relation pairs were most confusable with each other; however, it is the latter multi-class experiments that are most indicative of the real-world usefulness of rhetorical classication using prosodic features. Since our goal is to label meeting transcripts with rhetorical relations as a preprocessing step for automatic summarization, multi-class classification must be quite good to be at all useful.

5 Results

The following subsections give results on a development set of 175 relation pairs and on a test set of 75 relation pairs.

5.1 Development Set Results

5.1.1 Pairwise

The pairwise classification results on the development set are quite encouraging, showing that prosodic cues alone can yield an average of 68% classification success. Because equal class sizes were used in all data sets, the baseline classification would be 50%. The manually-labelled training data resulted in the highest accuracy, with the unsupervised technique performing slightly worse and the combination approach showing no added benefit to using manually-labelled data alone. Relation pairs involving the *question* relation generally perform the best, with the single highest pairwise classification being between *elaboration* and *question*. *Elaboration* is also generally discernible from *contrast* and *summary*.

	Cause	Contr.	Elab.	Q/A	Summ.
Cause	**15**	7	11	1	9
Contrast	8	**16**	9	6	5
Elaboration	6	4	**6**	2	4
Question	2	8	4	**17**	**10**
Summary	4	0	5	9	7
SUCCESS:	**34.8%**				

Table 3: Confusion Matrix for Development Set

Relation Pair	Super.	Unsuper.	Combo
Contrast/Cause	0.67	0.47	0.57
Contrast/Summary	0.60	0.43	0.50
Contrast/Question	0.70	0.73	0.77
Contrast/Elaboration	0.67	0.37	0.77
Cause/Summary	0.67	0.63	0.70
Cause/Question	0.87	0.77	0.80
Cause/Elaboration	0.47	0.57	0.50
Summary/Question	0.43	0.60	0.57
Summary/Elaboration	0.77	0.57	0.57
Elaboration/Question	0.80	0.60	0.57
AVERAGE:	**67%**	**58%**	**61%**

Table 4: Pairwise Results on Test Set

5.1.2 Multi-Class

The multi-class classification on the development set attained an accuracy of 0.35 using a one-against-the-rest classification approach, with chance level classification being 0.20. The confusion matrix in Table 3 illustrates the difficulty of multi-class classification; while *cause*, *contrast* and *question* relations are classified with considerable success, the *elaboration* relation pairs are often misclassified as *cause* and the *summary* pairs misclassifed as *question*.

5.2 Test Set Results

5.2.1 Pairwise

The pairwise results on the test set are similar to those of the development set, with the manually-labelled training set yielding superior results to the other two approaches, and relation pairs involving *question* and *elaboration* relations being particularly discernible. The average accuracy of the supervised approach applied to the test set is 67%, which closely mirrors the results on the development set. The most confusable pairs are *summary/question* and *cause/elaboration*; the former is quite surprising in that the *question* nucleus would be expected to have a prosody quite distinct from the others.

5.2.2 Multi-Class

The multi-class classification on the test set was considerably worse than the development set, with a success rate of only 0.24 (baseline: 0.2).

5.3 Features Analysis

This section details the prosodic characteristics of the *manually labelled* relations in the training, development, and test sets.

The *contrast* relation is typically realized with a low rate-of-speech for the nucleus and high rate-of-speech for the satellite, little or no pause between nucleus and satellite, a relatively flat overall F0 slope for the nucleus, and a satellite that increases in energy from the beginning to the end of the dialogue act. Of the manually labelled data sets, 74% of the examples are within a single speaker's turn.

The *cause* relation typically has a very high duration for the nucleus but a large amount of the nucleus containing silence. The slope of the nucleus is typically flat and the nuclear rate-of-speech is low. The satellite has a low rate-of-speech, a large amount of silence, a high maximum F0 and a high duration. There is typically a long duraton between nucleus and satellite and the speakers of the nucleus and the satellite are the same. Of the manually labelled data sets, nearly 94% of the examples are within a single speaker's turn.

The *elaboration* relation is often realized with a high nuclear duration, a high satellite duration, a long pause in-between and a low rate-of-speech for the satellite. The satellite typically has a high maximum F0 and the speakers of the nucleus and satellite are the same. 95% of the manually labelled examples occur within a single speaker's turn.

With the *summary* relation, the nucleus typically has a steep falling overall F0 while the nucleus has a rising overall F0. There is a short pause and a short duration for both nucleus and satellite. The rate-of-speech for the satellite is typically very high and there is little silence. 48% of the manually labelled examples occur within a single speaker's turn.

Finally, the *question* relation has a number of unique characteristics. The rate-of-speech of the nucleus is very high and there is very little silence. Surprisingly, these examples do not have canonical question intonation, instead having a low maximum F0 for the nucleus and a declining slope at the end of the nucleus. The overall F0 for the satellite steeply declines and there is a high standard deviation. The energy levels for the second and third quarters of the satellite are high compared with the average satellite energy and there is very little silence in the satellite as a whole. There is little or no pause between satellite and nucleus and both nucleus and satellite have relatively short durations. The maximum F0 for the satellite is typically low, and the speaker of the satellite is almost always different than the speaker of the nucleus - 99% of the time.

6 Conclusion

These experiments attempted to classify five rhetorical relations between dialogue acts in meeting speech using prosodic features. We primarily focused on pitch contour using numerous features of pitch slope and variance that intend to approximate the contour. In addition, we incorporated pause, energy, rate-of-speech and duration into our feature set. Using an unsupervised bootstrapping approach, we automatically labelled a large amount of training data and compared this approach to using a very small training set of manually labelled data. Whereas Marcu and Echihabi used such a bootstrapping approach to learn additional lexical information about relation pairs, we used the automatically labelled examples to learn the prosodic correlates of the relations. However, even a small amount of manually-labelled training data outperformed the unsupervised method, which is the same conclusion of Sporleder and Lascarides (Sporleder and Lascarides, to appear 2006), and a combined training method gave no additional benefit. One possible explanation for the poor performance of the bootstrapping approach is that some of the templates were inadvertently ambiguous, e.g., "I mean" can signal an *elaboration* or a *summary* and *which* can signal an *elaboration* or the beginning of a *question* relation. Furthermore, one possible drawback in employing this bootstrapping method is that there may be a complementary distribution between prosodic and lexical features. We are using explicit lexical cues to build an automatically labelled training set, but such explicitly cued relations may not be prosodically distinct. For example, a question that is sig-

nalled by "Who" or "What" may not have canonical question intonation since it is lexically signalled. This relates to a finding of Sporleder and Lascarides, who report that the unsupervised method of Marcu and Echihabi only generalizes well to relations that are already explicitly signalled, i.e. which could be found just by using the templates themselves.

The pairwise results were quite encouraging, with the supervised training approach yielding average accuracies of 68% on the development and test sets. This illustrates that prosody alone is quite indicative of certain rhetorical relations between dialogue acts. However, the multi-class classification performance was not far above chance levels. If this automatic rhetorical analysis is to aid an automatic summarizaton system, we will need to expand the prosodic database and perhaps couple this approach with a limited lexical/discourse approach in order to improve the multi-class classification accuracy. But most importantly, if even a small amount of training data leads to decent pairwise classification using only prosodic features, then greatly increasing the amount of manual annotation should provide considerable improvement.

7 Acknowledgements

Thanks to Mirella Lapata and Caroline Sporleder for valuable feedback. Thanks to two anonymous reviewers for helpful suggestions. This work was partly supported by the European Union 6th FWP IST Integrated Project AMI (Augmented Multiparty Interaction, FP6-506811, publication 177).

References

N. Asher and A. Lascarides. 2004. *Logics of Conversation*. Cambridge University Press, Cambridge, GB.

S. Corston-Oliver. 1998. *Computing representations of the structure of written discourse*. Ph.D. thesis, UC Santa Barbara.

H. den Ouden. 2004. *The Prosodic Realization of Text Structure*. Ph.D. thesis, University of Utrecht.

A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters. 2003. The ICSI meeting corpus. In *Proceedings of IEEE ICASSP 2003, Hong Kong, China*.

Alistair Knott and Robert Dale. 1996. Choosing a set of coherence relations for text-generation: a data-driven approach. In Giovanni Adorni and Michael Zock, editors, *Trends in natural language generation: an artificial intelligence perspective*, pages 47–67. Springer-Verlag, Berlin.

A. Knott and T. Sanders. 1998. The classifi cation of coherence relations and their linguistic markers: An exploration of two languages. *Journal of Pragmatics*, 30:135–175.

William C. Mann and Sandra A. Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

D. Marcu and A. Echihabi. 2002. An unsupervised approach to recognizing discourse relations. In *The Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA*.

D. Marcu. 1997. From discourse structures to text summaries. In *The Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization, Madrid, Spain*, pages 82–88.

D. Marcu. 1999. A decision-based approach to rhetorical parsing. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics, Maryland, USA*, pages 365–372.

D. Marcu. 2000. The rhetorical parsing of unrestricted texts: A surface-based approach. *Computational Linguistics*, 26(3):395–448.

J. Moore and M. Pollack. 1992. A problem for RST: The need for multi-level discourse analysis. *Computational Linguistics*, 18(4):537–544.

L. Noordman, I. Dassen, M. Swerts, and J. Terken. 1999. Prosodic markers of text structure. In K. Van Hoek, A. Kibrik, and L. Noordman, editors, *Discourse Studies in Cognitive Linguistics*, pages 133–149. John Benjamins Publications, Amsterdam, NL.

E. Shriberg, R. Dhillon, S. Bhagat, J. Ang, , and H. Carvey. - 2004. The ICSI meeting recorder dialog act (MRDA) corpus. In *Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue, Cambridge, MA, USA*, pages 97–100.

C. Sporleder and A. Lascarides. to appear, 2006. Using automatically labelled examples to classify rhetorical relations: An assessment. *Natural Language Engineering*.

M. Taboada. 2004. *Building Coherence and Cohesion: Task-Oriented Dialogue in English and Spanish*. John Benjamins Publications, Amsterdam, NL.

Off-Topic Detection in Conversational Telephone Speech

Robin Stewart and **Andrea Danyluk**
Department of Computer Science
Williams College
Williamstown, MA 01267
{06rss_2, andrea}@cs.williams.edu

Yang Liu
Department of Computer Science
UT Dallas
Richardson, TX 75080
yangl@hlt.utdallas.edu

Abstract

In a context where information retrieval is extended to spoken "documents" including conversations, it will be important to provide users with the ability to seek informational content, rather than socially motivated small talk that appears in many conversational sources. In this paper we present a preliminary study aimed at automatically identifying "irrelevance" in the domain of telephone conversations. We apply a standard machine learning algorithm to build a classifier that detects off-topic sections with better-than-chance accuracy and that begins to provide insight into the relative importance of features for identifying utterances as on topic or not.

1 Introduction

There is a growing need to index, search, summarize and otherwise process the increasing amount of available broadcast news, broadcast conversations, meetings, class lectures, and telephone conversations. While it is clear that users have wide ranging goals in the context of information retrieval, we assume that some will seek only credible information about a specific topic and will not be interested in the socially-motivated utterances which appear throughout most conversational sources. For these users, a search for information about weather should not return conversations containing small talk such as "Nice weather we've been having."

In this paper we investigate one approach for automatically identifying "irrelevance" in the domain of telephone conversations. Our initial data consist of conversations in which each utterance is labeled as being on topic or not. We apply inductive classifier learning algorithms to identify useful features and build classifiers to automatically label utterances.

We begin in Section 2 by hypothesizing features that might be useful for the identification of irrelevant regions, as indicated by research on the linguistics of conversational speech and, in particular, small talk. Next we present our data and discuss our annotation methodology. We follow this with a description of the complete set of features and machine learning algorithms investigated. Section 6 presents our results, including a comparison of the learned classifiers and an analysis of the relative utility of various features.

2 Linguistics of Conversational Speech

Cheepen (Cheepen, 1988) posits that speakers have two primary goals in conversation: **interactional** goals in which interpersonal motives such as social rank and trust are primary; and **transactional** goals which focus on communicating useful information or getting a job done. In a context where conversations are indexed and searched for information, we assume in this paper that users will be interested in the communicated information, rather than the way in which participants interact. Therefore, we assume that utterances with primarily transactional purposes will be most important, while interactional utterances can be ignored.

Greetings and partings are the most predictable

Proceedings of the Analyzing Conversations in Text and Speech (ACTS) Workshop at HLT-NAACL 2006, pages 8–14,
New York City, New York, June 2006.

type of interactional speech. They consistently appear at the beginning and end of conversations and follow a fairly formulaic pattern of content (Laver, 1981). Thus we hypothesize that: *Utterances near the beginning or end of conversations are less likely to be relevant.*

Cheepen also defines **speech-in-action** regions to be segments of conversation that are related to the present physical world or the activity of chatting, e.g. "What lovely weather." or "It is so nice to see you." Since these regions mainly involve participants identifying their shared social situation, they are not likely to contain transactional content. Further, since speech-in-action segments are distinguished by their focus on the present, we hypothesize that: *Utterances with present tense verbs are less likely to be relevant.*

Finally, small talk that is not intended to demarcate social hierarchy tends to be abbreviated, e.g. "Nice day" (Laver, 1981). From this we hypothesize that: *Utterances lacking common helper words such as "it", "there", and forms of "to be" are less likely to be relevant.*

3 Related Work

Three areas of related work in natural language processing have been particularly informative for our research.

First, speech act theory states that with each utterance, a conversant is committing an action, such as questioning, critiquing, or stating a fact. This is quite similar to the notion of transactional and interactional goals. However, speech acts are generally focused on the lower level of breaking apart utterances and understanding their purpose, whereas we are concerned here with a coarser-grained notion of relevance. Work closer to ours is that of Bates et al. (Bates et al., 2005), who define **meeting acts** for recorded meetings. Of their tags, **commentary** is most similar to our notion of relevance.

Second, there has been research on *generating* small talk in order to establish rapport between an automatic system and human user (Bickmore and Cassell, 2000). Our work complements this by potentially detecting off-topic speech from the human user as an indication that the system should also respond with interactional language.

Label	Utterance
S	2: [LAUGH] Hi.
S	2: How nice to meet you.
S	1: It is nice to meet you too.
M	2: We have a wonderful topic.
M	1: Yeah.
M	1: It's not too bad. [LAUGH]
T	2: Oh, I – I am one hundred percent in favor of, uh, computers in the classroom.
T	2: I think they're a marvelous tool, educational tool.

Table 1: A conversation fragment with annotations: (S)mall Talk, (M)etaconversation, and On-(T)opic. The two speakers are identified as "1" and "2".

Third, off-topic detection can be viewed as a segmentation of conversation into relevant and irrelevant parts. Thus our work has many similarities to topic segmentation systems, which incorporate cue words that indicate an abrupt change in topic (e.g. "so anyway..."), as well as long term variations in word occurrence statistics (Hearst, 1997; Reynar, 1999; Beeferman et al., 1999, e.g.). Our approach uses previous and subsequent sentences to approximate these ideas, but might benefit from a more explicitly segmentation-based strategy.

4 Data

In our work we use human-transcribed conversations from the Fisher data (LDC, 2004). In each conversation, participants have been given a topic to discuss for ten minutes. Despite this, participants often talk about subjects that are not at all related to the assigned topic. Therefore, a convenient way to define irrelevance in conversations in this domain is *segments which do not contribute to understanding the assigned topic*. This very natural definition makes the domain a good one for initial study; however, the idea can be readily extended to other domains. For example, broadcast debates, class lectures, and meetings usually have specific topics of discussion.

The primary transactional goal of participants in the telephone conversations is to discuss the assigned topic. Since this goal directly involves the act of discussion itself, it is not surprising that participants often talk about the current conversation or

the choice of topic. There are enough such segments that we assign them a special region type: **Metaconversation**. The purely irrelevant segments we call **Small Talk**, and the remaining segments are defined as **On-Topic**. We define utterances as segments of speech that are delineated by periods and/or speaker changes. An annotated excerpt is shown in Table 1.

For the experiments described in this paper, we selected 20 conversations: 4 from each of the topics "computers in education", "bioterrorism", "terrorism", "pets", and "censorship". These topics were chosen randomly from the 40 topics in the Fisher corpus, with the constraint that we wanted to include topics that could be a part of normal small talk (such as "pets") as well as topics which seem farther removed from small talk (such as "censorship").

Our selected data set consists of slightly more than 5,000 utterances. We had 2-3 human annotators label the utterances in each conversation, choosing from the 3 labels Metaconversation, Small Talk, and On-Topic. On average, pairs of annotators agreed with each other on 86% of utterances. The main source of annotator disagreement was between Small Talk and On-Topic regions; in most cases this resulted from differences in opinion of when exactly the conversation had drifted too far from the topic to be relevant.

For the 14% of utterances with mismatched labels, we chose the label that would be "safest" in the information retrieval context where small talk might get discarded. If any of the annotators thought a given utterance was On-Topic, we kept it On-Topic. If there was a disagreement between Metaconversation and Small Talk, we used Metaconversation. Thus, a Small Talk label was only placed if all annotators agreed on it.

Small Talk	Metaconv.	On-Topic
hi	topic	,
.	i	–
's	it	you
yeah	this	that
?	dollars	the
hello	so	and
oh	is	know
'm	what	a
in	was	wouldn
my	about	to
but	talk	like
name	for	his
how	me	they
we	okay	of
texas	do	't
there	phone	he
well	ah	uh
from	times	um
are	really	put
here	one	just

Table 2: The top 20 tokens for distinguishing each category, as ranked by the feature quality measure (Lewis and Gale, 1994).

5 Experimental Setup

5.1 Features

As indicated in Section 1, we apply machine learning algorithms to utterances extracted from telephone conversations in order to learn classifiers for Small Talk, Metaconversation, and On-Topic. We represent utterances as feature vectors, basing our selection of features on both linguistic insights and earlier text classification work. As described in Section 2, work on the linguistics of conversational speech (Cheepen, 1988; Laver, 1981) implies that the following features might be indicative of small talk: (1) position in the conversation, (2) the use of present-tense verbs, and (3) a lack of common helper words such as "it", "there", and forms of "to be".

To model the effect of proximity to the beginning of the conversation, we attach to each utterance a feature that describes its approximate position in the conversation. We do not include a feature for proximity to the end of the conversation because our transcriptions include only the first ten minutes of each recorded conversation.

In order to include features describing verb tense, we use Brill's part-of-speech tagger (Brill, 1992) . Each part of speech (POS) is taken to be a feature, whose value is a count of the number of occurrences in the given utterance.

To account for the words, we use a bag of words model with counts for each word. We normalize words from the human transcripts by converting everything to lower case and tokenizing contractions

Features	Values
n word tokens	for each word, # occurrences
standard POS tags as in Penn Treebank	for each tag, # occurrences
line number in conversation	0-4, 5-9, 10-19, 20-49, >49
utterance type	statement, question, fragment
utterance length (number of words)	1, 2, ..., 20, >20
number of laughs	laugh count
n word tokens in previous 5 utterances	for each word, total # occurrences in 5 previous
tags from POS tagger, previous 5	for each tag, total # occurrences in 5 previous
number of words, previous 5	total from 5 previous
number of laughs, previous 5	total from 5 previous
n word tokens, subsequent 5 utterances	for each word, total # occ in 5 subsequent
tags from POS tagger, subsequent 5	for each tag, total # occurrences in 5 subsequent
number of words, subsequent 5	total from 5 subsequent
number of laughs, subsequent 5	total from 5 subsequent

Table 3: Summary of features that describe each utterance.

and punctuation. We rank the utility of words according to the feature quality measure presented in (Lewis and Gale, 1994) because it was devised for the task of classifying similarly short fragments of text (news headlines), rather than long documents. We then consider the top n tokens as features, varying the number in different experiments. Table 2 shows the most useful tokens for distinguishing between the three categories according to this metric.

Additionally, we include as features the utterance type (statement, question, or fragment), number of words in the utterance, and number of laughs in the utterance.

Because utterances are long enough to classify individually but too short to classify reliably, we not only consider features of the current utterance, but also those of previous and subsequent utterances. More specifically, summed features are calculated for the five preceding utterances and for the five subsequent utterances. The number five was chosen empirically.

It is important to note that there is some overlap in features. For instance, the token "?" can be extracted as one of the n word tokens by Lewis and Gale's feature quality measure; it is also tagged by the POS tagger; and it is indicative of the utterance type, which is encoded as a separate feature as well. However, redundant features do not make up a significant percentage of the overall feature set.

Finally, we note that the conversation topic is *not* taken to be a feature, as we cannot assume that conversations in general will have such labels. The complete list of features, along with their possible values, is summarized in Table 3.

5.2 Experiments

We applied several classifier learning algorithms to our data: Naive Bayes, Support Vector Machines (SVMs), 1-nearest neighbor, and the C4.5 decision tree learning algorithm. We used the implementations in the Weka package of machine learning algorithms (Witten and Frank, 2005), running the algorithms with default settings. In each case, we performed 4-fold cross-validation, training on sets consisting of three of the conversations in each topic (15 conversations total) and testing on sets of the remaining 1 from each topic (5 total). Average training set size was approximately 3800 utterances, of which about 700 were Small Talk and 350 Metaconversation. The average test set size was 1270.

6 Results

6.1 Performance of a Learned Classifier

We evaluated the results of our experiments according to three criteria: accuracy, error cost, and plausibility of the annotations produced. In all

Algorithm	% Accuracy	Cohen's Kappa
SVM	**76.6**	**0.44**
C4.5	68.8	0.26
k-NN	64.1	0.20
Naive Bayes	58.9	0.27

Table 4: Classification accuracy and Cohen's Kappa statistic for each of the machine learning algorithms we tried, using all features at the 100-words level.

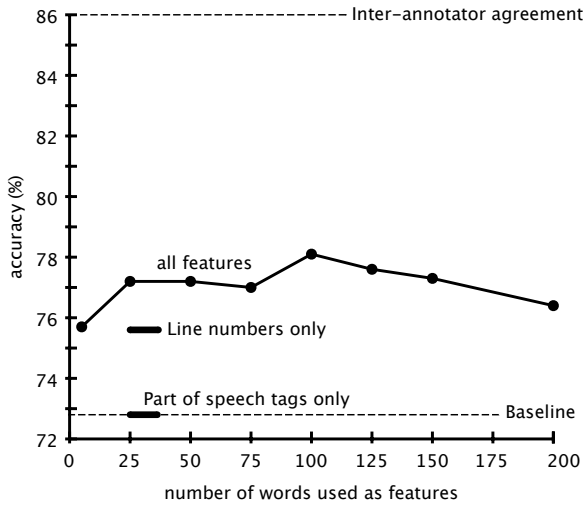


Figure 1: Classification results using SVMs with varying numbers of words.

cases our best results were obtained with the SVM. When evaluated on accuracy, the SVM models were the only ones that exceeded a baseline accuracy of 72.8%, which is the average percentage of On-Topic utterances in our data set. Table 4 displays the numerical results using each of the machine learning algorithms.

Figure 1 shows the average accuracy obtained with an SVM classifier using all features described in Section 5.1 except part-of-speech features (for reasons discussed below), and varying the number of words considered. While the best results were obtained at the 100-words level, all classifiers demonstrated significant improvement in accuracy over the baseline. The average standard deviation over the 4 cross-validation runs of the results shown is 6 percentage points.

From a practical perspective, accuracy alone is

S	M	T	<– classified as
55%	7%	38%	**S**mall Talk
21%	37%	42%	**M**etaconv.
8%	3%	89%	On **T**opic

Table 5: Confusion matrix for 100-word SVM classifier.

not an appropriate metric for evaluating our results. If the goal is to eliminate Small Talk regions from conversations, mislabeling On-Topic regions as Small Talk potentially results in the elimination of useful material. Table 5 shows a confusion matrix for an SVM classifier trained on a data set at the 100-word level. We can see that the classifier is conservative, identifying 55% of the Small Talk, but incorrectly labeling On-Topic utterances as Small Talk only 8% of the time.

Finally, we analyzed (by hand) the test data annotated by the classifiers. We found that, in general, the SVM classifiers annotated the conversations in a manner similar to the human annotators, transitioning from one label to another relatively infrequently as illustrated in Table 1. This is in contrast to the 1-nearest neighbor classifiers, which tended to annotate in a far more "jumpy" style.

6.2 Relative Utility of Features

Several of the features we used to describe our training and test examples were selected due to the claims of researchers such as Laver and Cheepen. We were interested in determining the relative contributions of these various linguistically-motivated features to our learned classifiers. Figure 1 and Table 6 report some of our findings. Using proximity to the beginning of the conversation ("line numbers") as a sole feature, the SVM classifier achieved an accuracy of 75.6%. This clearly verifies the hypothesis that utterances near the beginning of the conversation have different properties than those that follow.

On the contrary, when we used only POS tags to train the SVM classifier, it achieved an accuracy that falls exactly at the baseline. Moreover, removing POS features from the SVM classifier *improved* results (Table 6). This may indicate that detecting off-topic categories will require focusing on the words rather than the grammar of utterances. On

Condition	Accuracy	Kappa
All features	76.6	0.44
No word features	75.0	0.19
No line numbers	76.9	0.44
No POS features	77.8	0.46
No utterance type, length, or # laughs	76.9	0.45
No previous/next info	76.3	0.21
Only word features	**77.9**	**0.46**
Only line numbers	75.6	0.16
Only POS features	72.8	0.00
Only utterance type, length, and # laughs	74.1	0.09

Table 6: Percent accuracy and Cohen's Kappa statistic for the SVM at the 100-words level when features were left out or put in individually.

the other hand, part of speech information is implicit in the words (for example, an occurrence of "are" also indicates a present tense verb), so perhaps labeling POS tags does not add any new information. It is also possible that some other detection approach and/or richer syntactic information (such as parse trees) would be beneficial.

Finally, the words with the highest feature quality measure (Table 2) clearly refute most of the third linguistic prediction. Helper words like "it", "there", and "the" appear roughly evenly in each region type. Moreover, *all* of the verbs in the top 20 Small Talk list are forms of "to be" (some of them contracted as in "I'm"), while *no* "to be" words appear in the list for On-Topic. This is further evidence that differentiating off-topic speech depends deeply on the meaning of the words rather than on some more easily extracted feature.

7 Future Work

There are many ways in which we plan to expand upon this preliminary study. We are currently in the process of annotating more data and including additional conversation topics. Other future work includes:

- applying topic segmentation approaches to our data and comparing the results to those we have obtained so far;
- investigating alternate approaches for detecting Small Talk regions, such as smoothing with a Hidden Markov Model;
- using semi-supervised and active learning techniques to better utilize the large amount of unlabeled data;
- running the experiments with automatically generated (speech recognized) transcriptions, rather than the human-generated transcriptions that we have used to date. Our expectation is that such transcripts will contain more noise and thus pose new challenges;
- including prosodic information in the feature set.

Acknowledgements

The authors would like to thank Mary Harper, Brian Roark, Jeremy Kahn, Rebecca Bates, and Joe Cruz for providing invaluable advice and data. We would also like to thank the student volunteers at Williams who helped annotate the conversation transcripts, as well as the 2005 Johns Hopkins CLSP summer workshop, where this research idea was formulated.

References

Rebecca Bates, Patrick Menning, Elizabeth Willingham, and Chad Kuyper. 2005. Meeting Acts: A Labeling System for Group Interaction in Meetings. August.

Doug Beeferman, Adam Berger, and John Lafferty. 1999. Statistical Models for Text Segmentation. *Machine Learning*.

Timothy Bickmore and Justine Cassell. 2000. How about this weather?: Social Dialogue with Embodied Conversational Agents. AAAI Fall Symposium on Socially Intelligent Agents.

Eric Brill. 1992. A simple rule-based part of speech tagger. Proc. of the Third Conference on Applied NLP.

Christine Cheepen. 1988. *The Predictability of Informal Conversation*. Pinter Publishers, London.

Marti A. Hearst. 1997. TextTiling: Segmenting Text into Multiparagraph Subtopics Passages. *Computational Linguistics*, 23(1):33–64.

John Laver, 1981. *Conversational routine*, chapter Linguistic routines and politeness in greeting and parting, pages 289–304. Mouton, The Hague.

LDC. 2004. Fisher english training speech part 1, transcripts. http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2004T19.

David D. Lewis and William A. Gale. 1994. A sequential algorithm for training text classifiers. Proc. of SIGIR.

Jeffrey C. Reynar. 1999. Statistical models for topic segmentation. Proceedings of the 37th Annual Meeting of the ACL.

Ian H. Witten and Eibe Frank. 2005. *Data Mining: Practical Machine Learning Tools and Techniques, Second Edition*. Morgan Kaufmann, San Francisco.

Computational Measures for Language Similarity across Time in Online Communities

David Huffaker Joseph Jorgensen Francisco Iacobelli Paul Tepper Justine Cassell

Northwestern University

{d-huffaker, josephj, f-iacobelli, ptepper, justine}@northwestern.edu

Abstract

This paper examines language similarity in messages over time in an online community of adolescents from around the world using three computational measures: Spearman's Correlation Coefficient, Zipping and Latent Semantic Analysis. Results suggest that the participants' language *diverges* over a six-week period, and that divergence is not mediated by demographic variables such as leadership status or gender. This divergence may represent the introduction of more unique words over time, and is influenced by a continual change in subtopics over time, as well as community-wide historical events that introduce new vocabulary at later time periods. Our results highlight both the possibilities and shortcomings of using document similarity measures to assess convergence in language use.

1 Introduction

While document similarity has been a concern in computational linguistics for some time, less attention has been paid to change in similarity across time. And yet, while historical linguists have long addressed the issue of divergence or convergence among language groups over long periods of time, there has also been increasing interest in convergence (also referred to as entrainment, speech accommodation, or alignment) in other areas of Linguistics, with the realization that we have little understanding of change in very short periods of time, such as months, in a particular conversational setting, between two people, or in a large group.

The Internet provides an ideal opportunity to examine questions of this sort since all texts persevere for later analysis, and the diversity in kinds of online communities ensures that the influence of social behavior on language can be examined. Yet there has been very little work on language similarity in online communities.

In this paper we compare the use of three separate tools to measure document or message similarity in a large data set from an online community of over 3,000 participants from 140 different countries. Based on a review of related work on corpus similarity measures and document comparison techniques (Section 2.2), we chose Spearman's Correlation Coefficient, a comparison algorithm that utilizes GZIP (which we will refer to as "Zipping") and Latent Semantic Analysis. These three tools have all been shown effective for document comparison or corpus similarity, but never to our knowledge have any of them been used for document similarity over time, nor have they been compared to one another. Even though each of these tools is quite different in what it specifically measures and how it is used, and each has been used by quite different communities of researchers, they are all fairly well-understood (Section 4).

2 Related Work

In the next sections, we review literature on language similarity or convergence. We also review literature on the three computational tools, Spearman's Correlation Coefficient (SCC), Zipping, and Latent Semantic Analysis (LSA).

2.1 Language Similarity in Computer-mediated Communication

In dyadic settings, speakers often converge to one another's speech styles, not only matching the choice of referring expressions or other words, but also structural dimensions such as syntax, sound characteristics such as accent, prosody, or phonol-

Proceedings of the Analyzing Conversations in Text and Speech (ACTS) Workshop at HLT-NAACL 2006, pages 15–22,
New York City, New York, June 2006.

ogy, or even non-verbal behaviors such as gesture (Brennan & Clark, 1996; Street & Giles, 1982).

Some scholars suggest that this convergence or entrainment is based on a conscious need to accommodate to one's conversational partner, or as a strategy to maximize communication effectiveness (Street & Giles, 1982). Others suggest that the alignment is an automatic response, in which echoic aspects of speech, gesture and facial expressions are unconscious reactions (Garrod & Anderson, 1987; Lakin, Jefferies, Cheng, & Chartrand, 2003). In short, conversational partners tend to accommodate to each other by imitating or matching the semantic, syntactic and phonological characteristics of their partners (Brennan & Clark, 1996; Garrod & Pickering, 2004).

Many studies have concentrated on dyadic interactions, but large-scale communities also demonstrate language similarity or convergence. In fact, speech communities have a strong influence in creating and maintaining language patterns, including word choice or phonological characteristics (Labov, 2001). Language use often plays an important role in constituting a group or community identity (Eckert, 2003). For example, language 'norms' in a speech community often result in the conformity of new members in terms of accent or lexical choice (Milroy, 1980). This effect has been quite clear among non-native speakers, who quickly pick up the vernacular and speech patterns of their new situation (Chambers, 2001), but the opposite is also true, with native speakers picking up speech patterns from non-native speakers (Auer & Hinskens, 2005)

Linguistic innovation is particularly salient on the Internet, where words and linguistic patterns have been manipulated or reconstructed by individuals and quickly adopted by a critical mass of users (Crystal, 2001). Niederhoffer & Pennebaker (2002) found that users of instant messenger tend to match each other's linguistic styles. A study of language socialization in a bilingual chat room suggests that participants developed particular linguistic patterns and both native and non-native speakers were influenced by the other (Lam, 2004). Similar language socialization has been found in ethnographic research of large-scale online communities as well, in which various expressions are created and shared by group members (Baym, 2000; Cherny, 1999).

Other research not only confirms the creation of new linguistic patterns online, and subsequent adoption by users, but suggests that the strength of the social ties between participants influences how patterns are spread and adopted (Paolillo, 2001). However, little research has been devoted to how language changes over longer periods of time in these online communities.

2.2 Computational Measures of Language Similarity

The unit of analysis in online communities is the (e-mail or chat) message. Therefore, measuring entrainment in online communities relies on assessing whether or not similarity between the messages of each participant increases over time. Most techniques for measuring document similarity rely on the analysis of word frequencies and their co-occurrence in two or more corpora (Kilgarriff, 2001), so we start with these techniques.

Spearman's Rank Correlation Coefficient (SCC) is particularly useful because it is easy to compute and not dependent on text size. Unlike some other statistical approaches (e.g. chi-square), SCC has been shown effective on determining similarity between corpora of varying sizes, therefore SCC will serve as a baseline for comparison in this paper (Kilgarriff, 2001).

More recently, researchers have experimented with data compression algorithms as a measure of document complexity and similarity. This technique uses compression ratios as an approximation of a document's information entropy (Baronchelli, Caglioti, & Loreto, 2005; Benedetto, Caglioti, & Loreto, 2002). Standard Zipping algorithms have demonstrated effectiveness in a variety of document comparison and classification tasks. Behr et al. (2003) found that a document and its translation into another language compressed to approximately the same size. They suggest that this could be used as an automatic measure for testing machine translation quality. Kaltchenko (2004) argues that using compression algorithms to compute relative entropy is more relevant than using distances based on Kolmogorov complexity. Lastly, Bendetto et al. (2002) present some basic findings using GZIP for authorship attribution, determining the language of a document, and building a tree of language families from a text written in different languages. Although Zipping may be a conten-

tious technique, these results present intriguing reasons to continue exploration of its applications.

Latent Semantic Analysis is another technique used for measuring document similarity. LSA employs a vector-based model to capture the semantics of words by applying Singular Value Decomposition on a term-document matrix (Landauer, Foltz, & Laham, 1998). LSA has been successfully applied to tasks such as measuring semantic similarity among corpora of texts (Coccaro & Jurafsky, 1998), measuring cohesion (Foltz, Kintsch, & Landauer, 1998), assessing correctness of answers in tutoring systems (Wiemer-Hastings & Graesser, 2000) and dialogue act classification (Serafin & Di Eugenio, 2004).

To our knowledge, statistical measures like SCC, Zipping compression algorithms, or LSA have never been used to measure similarity of messages over time, nor have they been applied to online communities. However, it is not obvious how we would verify their performance, and given the nature of the task – similarity in over 15,000 e-mail messages – it is impossible to compare the computational methods to hand-coding. As a preliminary approach, we therefore decided to apply all three methods in turn to the messages in an online community to examine change in linguistic similarity over time, and to compare their results. Through the combination of lexical, phrasal and semantic similarity metrics, we hope to gain insight into the questions of whether entrainment occurs in online communities, and of what computational measures can be used to measure it.

2.3 The Junior Summit

The Junior Summit launched in 1998 as a closed online community for young people to discuss how to use technology to make the world better. 3000 children ages 10 to 16 participated in 1000 teams (some as individuals and some with friends). Participants came from 139 different countries, and could choose to write in any of 5 languages. After 2 weeks online, the young people divided into 20 topic groups of their own choosing. Each of these topic groups functioned as a smaller community within the community of the Junior Summit; after another 6 weeks, each topic group elected 5 delegates to come to the US for an in-person forum. The dataset from the Junior Summit comprises more than 40,000 e-mail messages; however, in the current paper we look at only a sub-set of these data – messages written *in English* during the 6-week *topic group* period. For complete details, please refer to Cassell & Tversky (2005).

3 The Current Study

In this paper, we examine entrainment among 419 of the 1000 user groups (the ones who wrote in English) and among the 15366 messages they wrote over a six-week period (with participants divided into 20 topic groups, with an average of 20.95 English writers per group). We ask whether the young people's language converges over time in an online community. Is similarity between the texts that are produced by the young people greater between adjacent weeks than between the less proximally-related weeks? Furthermore, what computational tools can effectively measure trends in similarity over time?

3.1 Hypotheses

In order to address these questions, we chose to examine change in similarity scores along two dimensions: (1) at the level of the individual; and (2) across the group as a whole. More specifically, we examine similarity between all pairs of individuals in a given topic group over time. We also compared similarity across the entire group at different time periods.

As depicted below, we first look at pairwise comparisons between the messages of participants in a particular topic group within a given time period, T_k (one week). For every pair of participants in a group, we calculated the similarity between two documents, each comprising all messages for a participant in the pair. Then we averaged the scores computed for all topic groups within a time period T_k and produced P_{Tk}, the average, pairwise similarity score for T_k. Our first hypothesis is that the average, pairwise similarity will increase over time, such that:

$$P_{T1} < P_{T2} < P_{T3} < P_{T4} < P_{T5} < P_{T6}$$

For our second set of tests, we compared all messages from a single time period to all messages of a previous time period within a single topic group. Our hypothesis was that temporal proximity would correlate with mean similarity, such that the messages of two adjacent time periods would exhibit more similarity than those of more distant

time periods. In order to examine this, we perform two individual hypothesis tests, where M_k is the document containing all the messages produced in time period T_k, and S(X,Y) is the similarity score for the two documents X and Y.

a) $S(M_k, M_{k-1}) > S(M_k, M_{k-2})$
b) $S(M_k, M_{k-1}) > S(M_k, M_1)$

Finally, we posit that SCC, Zipping and LSA will yield similar results for these tests.

4 Method

To prepare the data, we wrote a script to remove the parts of messages that could interfere with computing their similarity, in particular quoted messages and binary attachments, which are common in a corpus of email-like messages. We also removed punctuation and special characters.

4.1 Spearman's Correlation Coefficient

SCC is calculated as in Kilgarriff (2001). First, we compile a list of the common words between the two documents. The statistic can be calculated on the *n* most common words, or on all common words (i.e. *n* = total number of common words). We applied the latter approach, using all the words in common for each document pair. For each document, the *n* common words are ranked by frequency, with the lowest frequency word ranked 1 and the highest ranked *n*. For each common word, *d* is the difference in rank orders for the word in each document. SCC a normalized sum of the squared differences:

$$\rho = 1 - \frac{6\sum d^2}{n(n^2-1)}$$

The sum is taken over the *n* most frequent common words. In the case of ties in rank, where more than one word in a document occurs with the same frequency, the average of the ranks is assigned to the tying words. (For example, if words w_1, w_2 and w_3 are ranked 5th, 6th and 7th then all three words would be assigned the same rank of $\frac{5+6+7}{3} = 6$).

4.2 Zipping

When compressing a document, the resulting compression ratio provides an estimate of the document's entropy. Many compression algorithms generate a dictionary of sequences based on frequency that is used to compress the document. Likewise, one can leverage this technique to determine the similarity between two documents by assessing how optimal the dictionary generated when compressing one document is when applied to another document. We used GZIP for compression, which employs a combination of the LZ77 algorithm and Huffman coding. We based our approach on the algorithm used by (Benedetto, Caglioti, & Loreto, 2002), where the cross-entropy per character is defined as:

$$\frac{\text{length}(\text{zip}(A+B)) - \text{length}(\text{zip}(A))}{\text{length}(B)}$$

Here, *A* and *B* are documents; $A+B$ is document *B* appended to document *A*; zip(*A*) is the zipped document; and length(*A*) is the length of the document. It is important to note that the test document (B) needs to be small enough that it doesn't cause the dictionary to adapt to the appended piece. (Benedetto, Caglioti, & Loreto, 2002) refer to this threshold as the crossover length. The more similar the appended portion is, the more it will compress, and vice versa. We extended the basic algorithm to handle the extremely varied document sizes found in our data. Our algorithm does two one-way comparisons and returns the mean score. Each one-way comparison between two documents, *A* and *B*, is computed by splitting *B* into 300 character chunks. Then for each chunk, we calculated the cross entropy per character when appending the chunk onto *A*. Each one-way comparison returns the mean calculation for every chunk.

We fine-tuned the window size with a small, hand-built corpus of news articles. The differences are slightly more pronounced with larger window sizes, but that trend starts to taper off between window sizes of 300 and 500 characters. In the end we chose 300 as our window size, because it provided sufficient contrast and yet still gave a few samples from even the smallest documents in our primary corpus.

4.3 Latent Semantic Analysis (LSA)

For a third approach, we used LSA to analyze the semantic similarity between messages across different periods of time. We explored three imple-

mentations of LSA: (a) the traditional algorithm described by Foltz et al (1998) with one semantic space per topic group, (b) the same algorithm but with one semantic space for all topic groups and (c) an implementation based on *Word Space* (Schutze, 1993) called *Infomap*. All three were tested with several settings such as variations in the number of dimensions and levels of control for stop words, and all three demonstrated similar results. For this paper, we present the Infomap results due to its wide acceptance among scholars as a successful implementation of LSA.

To account for nuances of the lexicon used in the Junior Summit data, we built a semantic space from a subset of this data comprised of 7000 small messages (under one kb) and 100 dimensions without removing stop words. We then built vectors for each document and compared them using cosine similarity (Landauer, Foltz, & Laham, 1998).

5 Results

The tools we employ approach document similarity quite differently; we therefore compare findings as a way of triangulating on the nature of entrainment in the Junior Summit online community.

5.1 Pairwise Comparisons over Time

First, we hypothesized that messages between individuals in a given topic group would demonstrate more similarity over time. Our findings did not support this claim; in fact, they show the opposite. All three tests show slight convergence between time period one and two, some variation, and then divergence between time periods four, five and six.

Spearman's Correlation Coefficient demonstrates a steady decline in similarity. As shown in Figure 1, the differences between time periods were all significant, $F_{(5,1375)} = 21.475$, $p<.001$, where $N=1381$ (N represents user pairs across all six time periods).

Zipping also shows a significant difference between each time period, $F_{(5,1190)} = 39.027$, $p<.001$, $N=1196$, demonstrating a similar decline in similarity, although not as unwavering. See Figure 2.

LSA demonstrates the same divergent trend over time, $F_{(5,1410)} = 27.139$, $p<.001$, $N=1416$, with a slight spike at T_4 and T_5. While the dip at time 3 is more pronounced than SCC and Zipping, it is still consistent with the overall findings of the other measures. See Figure 3.

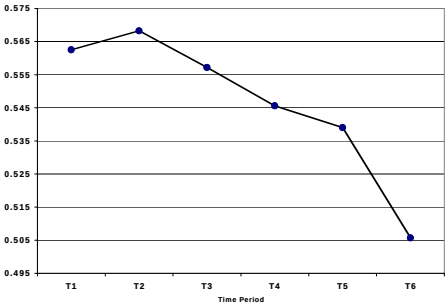


Figure 1. Spearman's Correlation Coefficient Similarity Scores for all Pairwise comparisons, $T_1 - T_6$

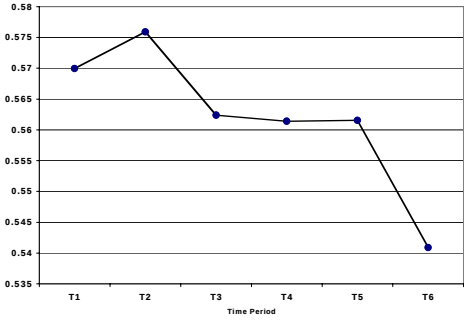


Figure 2. Zipping Similarity Scores for all Pairwise comparisons, $T_1 - T_6$

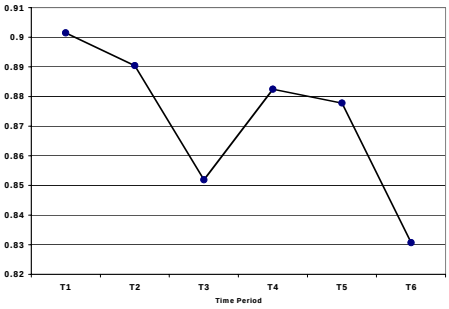


Figure 3. LSA Similarity Scores for all Pairwise comparisons, $T_1 - T_6$.

Because of these surprising findings, we examined the influence of demographic variables, such as leadership (those chosen as delegates from each topic group to the in-person forum), gender, and the particular topic groups the individuals were a part of. We divided delegate pairs into (a) pairs where both individuals are delegates; (b) pairs where both individuals are non-delegates; and (c) mixed pairs of delegates and non-delegates. Similarly, gender pairs were divided into same-sex (e.g., male-male, female-female) and mixed-sex

pairs. For topic groups, we re-ran our analyses on each of the 20 topic groups separately.

Overall, both leaders and gender pairs demonstrate the same divergent trends as the group as a whole. However, not all tests showed significant differences when comparing these pairs.

For instance, Spearman's Correlation Coefficient found a significant difference in similarity between three groups, where $F_{(2,273)} = 6.804$, $p<.001$, $n=276$, such that delegate-delegate pairs demonstrate higher similarity scores than non-delegate pairs and mixed pairs. LSA found the same result, $F(2,280) = 11.122$, $p<.001$ $n=283$. By contrast, Zipping did not find this to be the case, where $F_{(2,226)} = 2.568$, $p=.079$, $n=229$.

In terms of the potential effect of gender on similarity scores, Zipping showed a significant difference between the three groups, $F_{(2,236)} = 3.546$, $p<.05$, $n=239$, such that female-female pairs and mixed-sex pairs demonstrate more similarity than male-male pairs. LSA found the same relationship, $F_{(2,280)} = 4.79$, $p<.005$ $n=283$. By contrast, Spearman's Correlation Coefficient does not show a significant between-groups difference, $F_{(2,273)} = .699$, $p=.498$, $n=276$.

In terms of differences among the topic groups, we did indeed find differences such that some topic groups demonstrated the fairly linear slope with decreasingly similarity shown above, while others demonstrated dips and rises resulting in a level of similarity at T6 quite similar to T1. There is no neat way to statistically measure the differences in these slopes, but it does indicate that future analyses need to take topic group into account.

In sum, we did not find leadership or gender to mediate language similarity in this community. Topic group, on the other hand, did play a role, however no topic groups showed increasing *similarity* across time.

5.2 Similarity and Temporal Proximity

Our second hypothesis concerned the gradual change of language over time such that temporal proximity of time periods would correlate with mean similarity. In other words, we expect that messages in close time periods (e.g., adjacent weeks) should be more similar than messages from more distant time periods. In order to examine this, we performed two individual tests, in which our predictions can be described as follows: (a) the similarity between texts in one time period and texts in the neighboring time period is greater than texts in one time period, and texts that came two periods previously, $S(M_k, M_{k-1}) > S(M_k, M_{k-2})$; and (b) the similarity between texts in one time period and texts in the neighboring time period is greater than the similarity between texts in one time period, and texts in the very first time period, $S(M_k, M_{k-1}) > S(M_k, M_1)$.

As shown in Table 1, SCC and Zipping tests confirm these hypotheses, while none of the LSA tests revealed significant differences.

Table 1. Temporal Proximity Similarities SCC, Zipping, and LSA, $n=20$ topic groups

	$S(M_k,M_{k-1}) > S(M_k,M_{k-2})$	$S(M_k,M_{k-1}) > S(M_k,M_1)$	$S(M_k,M_{k-2}) > S(M_k,M_1)$
SCC	.665 > .653†	.665 > .639°	.653 > .639°
ZIP	.628 > .608†	.628 > .605†	.608 > .605§
LSA	9.74 > .971	9.74 > .971	.97166 < .97168

Note: *$p<.05$, °$p<.01$, †$p<.001$, §$p = .0525$, one-tailed

6 Discussion

This work presents several novel contributions to the analysis of text-based messages in online communities. Using three separate tools, Spearman's Correlation Coefficient, Zipping and Latent Semantic Analysis measures, we found that across time, members of an online community diverge in the language they use. More specifically, a comparison of the words contributed by any pair of users in a particular topic group shows increasing *dissimilarity* over the six-week period.

This finding seems counter-intuitive given work in linguistics and psychology, which shows that dyads and communities converge, entrain and echo each other's lexical choices and communication styles. Similarly, our own temporal proximity results appear to indicate convergence, since closer time periods are more similar than more distant ones. Finally, previous hand-coding of these data revealed convergence, for example between boys and girls on the use of emotion words, between older and younger children on talk about the future (Cassell & Tversky, 2005). So we ask, why do our tools demonstrate this divergent trend?

We believe that one answer comes from the fact that, while the young people may be discussing a more restricted range of topics, they are contributing a wider variety of vocabulary. In order to examine whether indeed there were more unique

words over time, we first simply manually compared the frequency of words over time and found that, on the contrary, there are consistently fewer unique words by T_6, which suggests convergence. However, there are also fewer and fewer *total* words by the end of the forum. This is due to the number of participants who left the forum after they were not elected to go to Boston. If we divide the unique words by the total words, we find that the *ratio* of unique words consistently increases over time (see Figure 4). It is likely that this ratio contributes to our results of divergence.

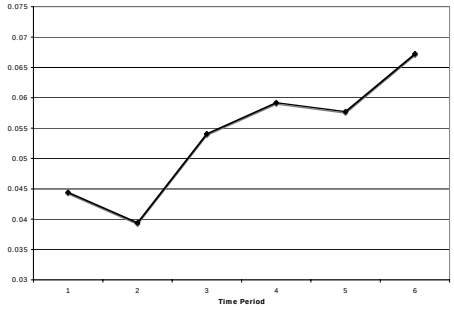


Figure 4. Ratio of Unique to Total Words, $T_1 - T_6$

In order to further examine the role of increasing vocabulary in the Junior Summit as a whole, we also created several control groups comprised of random pairs of users (i.e., users that had never written to each other), and measured their pairwise similarity across time. The results were similar to the experimental groups, demonstrating a slope with roughly the same shape. This argues for convergence and divergence being affected by something at a broader, community-level such as an increase in vocabulary.

This result is interesting for an additional reason. Some users – perhaps particularly non-native speakers or younger adolescents, may be learning new vocabulary from other speakers, which they begin to introduce at later time periods. An increasingly diversified vocabulary could conceivably result in differences in word frequency among speakers. This leads us to some key questions: to what extent does the language of individuals change over time? Is individual language influenced by the language of the community? This is heart of entrainment.

In conclusion, we have shown that SCC, Zipping and LSA can be used to assess message similarity over time, although they may be somewhat blunt instruments for our purposes. In addition, while Zipping is somewhat contentious and not as widely-accepted as SCC or LSA is, we found that the three tools provide very similar results. This is particularly interesting given that, while all three methods take into account word or word-sequence frequencies, LSA is designed to also take into account aspects of semantics beyond the surface level of lexical form.

All in all, these tools not only contribute to ways of measuring similarity across documents, but can be utilized in measuring smaller texts, such as online messages or emails. Most importantly, these tools remind us how complex and dynamic everyday language really is, and how much this complexity must be taken into account when building computational tools for the analysis of text and conversation.

6.1 Future Directions

In future work, we intend to find ways to compare the results obtained from different topic groups and also to examine differences among individual users, including re-running our analyses after removing outliers. We also hope to explore the interplay between individuals and the community and changes in language similarity. In other words, can we find those individuals who may be acquiring new vocabulary? Are there "language leaders" responsible for language change online?

We also plan to analyze words in terms of their local contexts, to see if this changes over time and how it impacts our results. Furthermore, we intend to go beyond word frequency to classify topic changes over time to get a better understanding of the dynamics of the groups (Kaufmann, 1999).

Finally, as we have done in the past with our analyses of this dataset, we would like to perform a percentage of hand-coded, human content analysis to check reliability of these statistical methods.

Acknowledgements

Thanks to members of the Articulab, Stefan Kaufmann, Stefan Wuchty, Will Thompson, Debbie Zutty and Lauren Olson for invaluable input. This research was in part supported by a generous grant from the Kellogg Foundation.

References

Auer, P., & Hinskens, F. (2005). The role of interpersonal accommodation in a theory of language change. In P. Auer, F. Hinskens & P. Kerswill

(Eds.), *Dialect change: The convergence and divergence of dialects in European languages* (pp. 335-357). Cambridge, MA: Cambridge University Press.

Baronchelli, A., Caglioti, E., & Loreto, V. (2005). Artificial sequences and complexity measures. *Journal of Statistical Mechanics: Theory and Experiment, P04002*, 1-26.

Baym, N. K. (2000). *Tune in, log on: Soaps, fandom, and online community*. New York: Sage Publications.

Benedetto, D., Caglioti, E., & Loreto, V. (2002). Language trees and zipping. *Physical Review Letters, 88*(4), 1-4.

Brennan, S. E., & Clark, H. H. (1996). Conceptual pacts and lexical choice in conversation. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 22*(6), 1482-1493.

Cassell, J., & Tversky, D. (2005). The language of online intercultural community formation. *Journal of Computer-Mediated Communication, 10*(2), Article 2.

Chambers, J. K. (2001). Dynamics of dialect convergence. *Journal of Sociolinguistics, 6*(1), 117-130.

Cherny, L. (1999). *Conversation and Community: Chat in a Virtual World*. Stanford: Center for the Study of Language and Information.

Coccaro, N., & Jurafsky, D. (1998, November 1998). *Towards better integration of semantic predictors in statistical language modeling.* Paper presented at the International Conference on Spoken Language Processing (ICSLP-98), Sidney, Australia.

Crystal, D. (2001). *Language and the Internet*. New York: Cambridge University Press.

Eckert, P. (2003). Language and adolescent peer groups. *Journal of Language and Social Psychology, 22*(1), 112-118.

Foltz, P. W., Kintsch, W., & Landauer, T. K. (1998). The measurement of textual Coherence with Latent Semantic Analysis. *Discourse Processes, 25*, 285-307.

Garrod, S., & Anderson, A. (1987). Saying what you mean in dialogue: A study in conceptual and semantic coordination. *Cognition, 27*, 181-218.

Garrod, S., & Pickering, M. J. (2004). Why is conversation so easy? *Trends in Cognitive Sciences, 8*(1), 8-11.

Kalthchenko, A. (2004, May 2-5, 2004). *Algorithms for estimation of information distance with application to bioinformatics and linguistics.* Paper presented at the Canadian Conference on Electrical and Computer Engineering (CCECE 2004), Niagara Falls, Ontario, Canada.

Kaufmann, S. (1999). *Cohesion and collocation: Using context vectors in text segmentation.* Paper presented at the 37th Annual Meeting of the Association for Computational Linguistics, College Park, MD.

Kilgarriff, A. (2001). Comparing corpora. *International Journal of Corpus Linguistics, 6*(1), 97-133.

Labov, W. (2001). *Principles of linguistic change* (Vol. 2: Social Factors). Oxford: Blackwell Publishers.

Lakin, J. L., Jefferies, V. E., Cheng, C. M., & Chartrand, T. L. (2003). The chameleon effect as social glue: Evidence for the evolutionary significance of nonconscious mimicry. *Journal of Nonverbal Behavior, 27*(3), 145-162.

Lam, W. S. E. (2004). Second language socialization in a bilingual chat room: Global and local considerations. *Language Learning & Technology, 8*(3), 44-65.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). Introduction to latent semantic analysis. *Discourse Processes, 25*, 259-284.

Milroy, L. (1980). *Language and social networks*. Oxford: Blackwell Publishers.

Niederhoffer, K. G., & Pennebaker, J. W. (2002). Linguistic style matching in social interaction. *Journal of Language and Social Psychology, 21*(4), 337-360.

Paolillo, J. (2001). Language variation on internet relay chat: A social network approach. *Journal of Sociolinguistics, 5*(2), 180-213.

Schutze, H. (1993). Word space. In S. J. Hanson, J. D. Cowan & C. L. Giles (Eds.), *Advances in Neural Information Processing Systems 5*. San Mateo, CA: Morgan Kaufmann Publishers.

Serafin, R., & Di Eugenio, B. (2004, July 21-26, 2004). *FLSA: Extending latent semantic analysis with features for dialogue act classification.* Paper presented at the 42nd Annual Meeting for the Association of Computational Linguistics (ACL04), Barcelona, Spain.

Street, R. L., & Giles, H. (1982). Speech accommodation theory. In M. E. Roloff & C. R. Berger (Eds.), *Social cognition and communication* (pp. 193-226). London: Sage Publications.

Wiemer-Hastings, P., & Graesser, A. C. (2000). Select-a-Kibitzer: A computer tool that gives meaningful feedback on student compositions. *Interactive Learning Environments, 8*(2), 149–169.

You Are What You Say: Using Meeting Participants' Speech to Detect their Roles and Expertise

Satanjeev Banerjee
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA 15213
banerjee@cs.cmu.edu

Alexander I. Rudnicky
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
air@cs.cmu.edu

Abstract

Our goal is to automatically detect the functional roles that meeting participants play, as well as the expertise they bring to meetings. To perform this task, we build decision tree classifiers that use a combination of simple speech features (speech lengths and spoken keywords) extracted from the participants' speech in meetings. We show that this algorithm results in a role detection accuracy of 83% on unseen test data, where the random baseline is 33.3%. We also introduce a simple aggregation mechanism that combines evidence of the participants' expertise from multiple meetings. We show that this aggregation mechanism improves the role detection accuracy from 66.7% (when aggregating over a single meeting) to 83% (when aggregating over 5 meetings).

1 Introduction

A multitude of meetings are organized every day around the world to discuss and exchange important information, to make decisions and to collaboratively solve problems. Our goal is to create systems that automatically understand the discussions at meetings, and use this understanding to assist meeting participants in various tasks during and after meetings. One such task is the retrieval of information from previous meetings, which is typically a difficult and time consuming task for the human to perform (Banerjee et al., 2005). Another task is to automatically record the action items being discussed at meetings, along with details such as when the action is due, who is responsible for it, etc.

Meeting analysis is a quickly growing field of study. In recent years, research has focussed on automatic speech recognition in meetings (Stolcke et al., 2004; Metze et al., 2004; Hain et al., 2005), activity recognition (Rybski and Veloso, 2004), automatic meeting summarization (Murray et al., 2005), meeting phase detection (Banerjee and Rudnicky, 2004) and topic detection (Galley et al., 2003). Relatively little research has been performed on automatically detecting the *roles* that meeting participants play as they participate in meetings. These roles can be functional (e.g. the *facilitator* who runs the meeting, and the *scribe* who is the designated note taker at the meeting), discourse based (e.g. the *presenter*, and the *discussion participant*), and expertise related (e.g. the *hardware acquisition expert* and the *speech recognition research expert*). Some roles are tightly scoped, relevant to just one meeting or even a part of a meeting. For example, a person can be the facilitator of one meeting and the scribe of another, or the same person can be a presenter for one part of the meeting and a discussion participant for another part. On the other hand, some roles have a broader scope and last for the duration of a project. Thus a single person may be the speech recognition expert in a project and have that role in all meetings on that project. Additionally, the same person can play multiple roles, e.g. the scribe can be a speech recognition expert too.

Automatic role detection has many benefits, espe-

Proceedings of the Analyzing Conversations in Text and Speech (ACTS) Workshop at HLT-NAACL 2006, pages 23–30,
New York City, New York, June 2006.

cially when used as a source of constraint for other meeting understanding components. For example, detecting the facilitator of the meeting might help the automatic topic detection module if we know that facilitators officially change topics and move the discussion from one agenda item to the next. Knowing who the speech recognition expert is can help the automatic action item detector: If an action item regarding speech recognition has been detected but the *responsible person* field has not been detected, the module may place a higher probability on the speech recognition expert as being the responsible person for that action item. Additionally, detecting who is an expert in which field can have benefits of its own. For example, it can be used to automatically direct queries on a particular subject to the person deemed most qualified to answer the question, etc. Basic information such as participant role and expertise needs to be robustly extracted if it is to be of use to the more sophisticated stages of understanding. Accordingly, we have based our role detection algorithm on simple and highly accurate speech features, as described in section 5.1.2.

(Banerjee and Rudnicky, 2004) describes the automatic detection of discourse roles in meetings. These roles included *presenter* (participants who make formal presentations using either slides or the whiteboard), *discussion participant* (participants involved in a discussion marked by frequent turn changes), *observer* (participants not speaking, but nevertheless consuming information during a presentation or discussion), etc. In this paper we focus on automatically detecting the *functional* and *expertise* based roles that participants play in a meeting. In the next section we describe the data that is used in all our role detection work in this paper. In subsequent sections we describe the role detection algorithm in more detail, and present evaluation results.

2 The Y2 Meeting Scenario Data

Our research work is part of the Cognitive Assistant that Learns and Organizes project (CALO, 2003). A goal of this project is to create an artificial assistant that can understand meetings and use this understanding to assist meeting participants during and after meetings. Towards this goal, data is being collected by creating a rich multimodal record of meetings (e.g. (Banerjee et al., 2004)). While a large part of this data consists of natural meetings (that would have taken place even if they weren't being recorded), a small subset of this data is "scenario driven" – the *Y2 Scenario Data*.

Meeting #	Typical scenario
1	Hiring Joe: Buy a computer and find office space for him
2	Hiring Cindy and Fred: Buy com–puters & find office space for them
3	Buy printer for Joe, Cindy and Fred
4	Buy a server machine for Joe, Cindy and Fred
5	Buy desktop and printer for the meeting leader

Table 1: Typical Scenario Instructions

The Y2 Scenario Data consists of meetings between groups of 3 or 4 participants. Each group participated in a sequence of up to 5 meetings. Each sequence had an overall scenario – the purchasing of computing hardware and the allocation of office space for three newly hired employees. Participants were told to assume that the meetings in the sequence were being held one week apart, and that between any two meetings "progress" was made on the action items decided at each meeting. Participants were given latitude to come up with their own stories of what "progress" was made between meetings. At each meeting, participants were asked to review progress since the last meeting and make changes to their decisions if necessary. Additionally, an extra topic was introduced at each meeting, as shown in table 1.

In each group of participants, one participant played the role of the *manager* who has control over the funds and makes the final decisions on the purchases. The remaining 2 or 3 participants played the roles of either the *hardware acquisition expert* or the *building facilities expert*. The role of the hardware expert was to make recommendations on the buying of computers and printers, and to actually make the purchases once a decision was made to do so. Similarly the role of the building expert was to make recommendations on which rooms were available to fit the new employees into. Despite this role assign-

ment, all participants were expected to contribute to discussions on all topics.

To make the meetings as natural as possible, the participants were given control over the evolution of the story, and were also encouraged to create conflicts between the manager's demands and the advice that the experts gave him. For example, managers sometimes requested that all three employees be put in a single office, but the facilities expert announced that no 3 person room was available, unless the manager was agreeable to pay extra for them. These conflicts led to extended negotiations between the participants. To promote fluency, participants were instructed to use their knowledge of existing facilities and equipment instead of inventing a completely fictitious set of details (such as room numbers).

The data we use in this paper consists of 8 sequences recorded at Carnegie Mellon University and at SRI International between 2004 and 2005. One of these sequences has 4 meetings, the remaining have 5 meetings each, for a total of 39 meetings. 4 of these sequences had a total of 3 participants each; the remaining 4 sequences had a total of 4 participants each. On average each meeting was 15 minutes long. We partitioned this data into two roughly equal sets, the training set containing 4 meeting sequences, and the test set containing the remaining 4 sets. Although a few participants participated in multiple meetings, there was no overlap of participants between the training and the test set.

3 Functional Roles

Meeting participants have **functional roles** that ensure the smooth conduct of the meeting, without regard to the specific contents of the meeting. These roles may include that of the *meeting leader* whose functions typically include starting the meeting, establishing the agenda (perhaps in consultation with the other participants), making sure the discussions remain on–agenda, moving the discussion from agenda item to agenda item, etc. Another possible functional role is that of a the designated *meeting scribe*. Such a person may be tasked with the job of taking the official notes or minutes for the meeting.

Currently we are attempting to automatically detect the meeting leader for a given meeting. In our data (as described in section 2) the participant playing the role of the *manager* is always the meeting leader. In section 5 we describe our methodology for automatically detecting the meeting leader.

4 Expertise

Typically each participant in a meeting makes contributions to the discussions at the meeting (and to the project or organization in general) based on their own expertise or skill set. For example, a project to build a multi–modal note taking application may include project members with expertise in speech recognition, in video analysis, etc. We define **expertise based roles** as roles based on skills that are relevant to participants' contributions to the meeting discussions and the project or organization in general. Note that the expertise role a participant plays in a meeting is potentially dependent on the expertise roles of the other participants in the meeting, and that a single person may play different expertise roles in different meetings, or even within a single meeting. For example, a single person may be the "speech recognition expert" on the note taking application project that simply uses off–the–shelf speech recognition tools to perform note taking, but a "noise cancellation" expert on the project that is attempting to improve the in–house speech recognizer. Automatically detecting each participant's roles can help such meeting understanding components as the action item detector.

Ideally we would like to automatically discover the roles that each participant plays, and cluster these roles into groups of similar roles so that the meeting understanding components can transfer what they learn about particular participants to other (and newer) participants with similar roles. Such a role detection mechanism would need no prior training data about the specific roles that participants play in a new organization or project. Currently however, we have started with a simplified participant role detection task where we do have training data pertinent to the specific roles that meeting participants play in the test set of meetings. As mentioned in section 2, our data consists of people playing two kinds of expertise–based roles – that of a hardware acquisition expert, and that of a building facilities expert. In the next section we discuss our

methodology of automatically detecting these roles from the meeting participants' speech.

5 Methodology

Given a sequence of longitudinal meetings, we define our role detection task as a three–way classification problem, where the input to the classifier consists of features extracted from the speech of a particular participant over the given meetings, and the output is a probability distribution over the three possible roles. Note that although a single participant can simultaneously play both a functional and an expertise–based role, in the Y2 Scenario Data each participant plays exactly one of the three roles. We take advantage of this situation to simplify the problem to the three way classification defined above. We induce a decision tree (Quinlan, 1986) classifier from hand labeled data. In the next subsection we describe the steps involved in training the decision tree role classifier, and in the subsequent subsection we describe how the trained decision tree is used to arrive at a role label for each meeting participant.

5.1 Training

5.1.1 Keyword List Creation

One of the sources of information that we wish to employ to perform functional and expertise role detection is the words that are spoken by each participant over the course of the meetings. Our approach to harness this information source is to use labeled training data to first create a set of words most strongly associated with each of the three roles, and then use only these words during the feature extraction phase to detect each participant's role, as described in section 5.1.2.

We created this list of keywords as follows. Given a training set of meeting sequences, we aggregated for each role all the speech from all the participants who had played that role in the training set. We then split this data into individual words and removed *stop words* – closed class words (mainly articles and prepositions) that typically contain less information pertinent to the task than do nouns and verbs. For all words across all the three roles, we computed the degree of association between each word and each of the three roles, using the chi squared method (Yang and Pedersen, 1997), and chose the top 200 high scoring word–role pairs. Finally we manually examined this list of words, and removed additional words that we deemed to not be relevant to the task (essentially identifying a domain–specific stop list). This reduced the list to a total of 180 words. The 5 most frequently occurring words in this list are: *computer, right, need, week* and *space*. Intuitively the goal of this keyword selection pre–processing step is to save the decision tree role classifier from having to automatically detect the important words from a much larger set of words, which would require more data to train.

5.1.2 Feature Extraction

The input to the decision tree role classifier is a set of features abstracted from a specific participant's speech. One strategy is to extract exactly one set of features from all the speech belonging to a participant across all the meetings in the meeting sequence. However, this approach requires a very large number of meetings to train. Our chosen strategy is to *sample* the speech output by each participant multiple times over the course of the meeting sequence, classify each such sample, and then aggregate the evidence over all the samples to arrive at the overall likelihood that a participant is playing a certain role.

To perform the sampling, we split each meeting in the meeting sequence into a sequence of contiguous windows each n seconds long, and then compute one set of features from each participant's speech during each window. The value of n is decided through parametric tests (described in section 7.1). If a particular participant was silent during the entire duration of a particular window, then features are extracted from that silence.

Note that in the above formulation, there is no overlap (nor gap) between successive windows. In a separate set of experiments we used *overlapping* windows. That is, given a window size, we moved the window by a fixed step size (less than the size of the window) and computed features from each such overlapping window. The results of these experiments were no better than those with non–overlapping windows, and so for the rest of this paper we simply report on the results with the non–overlapping windows.

Given a particular window of speech of a partic-

ular participant, we extract the following 2 speech *length* based features:

- Rank of this participant (among this meeting's participants) in terms of the length of his speech during this window. Thus, if this participant spoke the longest during the window, he has a feature value of 1, if he spoke for the second longest number of times, he has a feature value of 2, etc.

- Ratio of the length of speech of this participant in this window to the total length of speech from all participants in this window. Thus if a participant spoke for 3 seconds, and the total length of speech from all participants in this window was 6 seconds, his feature value is 0.5. Together with the rank feature above, these two features capture the amount of speech contributed by each participant to the window, relative to the other participants.

In addition, for each window of speech of a particular participant, and for each keyword in our list of pre–decided keywords, we extract the following 2 features:

- Rank of this participant (among this meeting's participants) in terms of the number of times this keyword was spoken. Thus if in this window of time, this participant spoke the keyword *printer* more often than any of the other participants, then his feature value for this keyword is 1.

- Ratio of the number of times this participant uttered this keyword in this window to the total number of times this keyword was uttered by all the participants during this window. Thus if a participant spoke the word *printer* 5 times in this window, and in total all participants said the word *printer* 7 times, then his feature value for this keyword is $5/7$. Together with the keyword rank feature above, these two features capture the number of times each participant utters each keyword, relative to the other participants.

Thus for each participant, for each meeting window, we extract two features based on the lengths of speech, and 2×180 features for each of the 180 keywords, for a total of 362 features. The true output label for each such data point is the role of that participant in the meeting sequence. We used these data points to induce a classifier using the Weka Java implementation (Witten and Frank, 2000) of the C4.5 decision tree learning algorithm (Quinlan, 1986). This classifier takes features as described above as input, and outputs class membership probabilities, where the classes are the three roles. Note that for the experiments in this paper we extract these features from the *manual transcriptions* of the speech of the meeting participants. In the future we plan to perform these experiments using the transcriptions output by an automatic speech recognizer.

5.2 Detecting Roles in Unseen Data

5.2.1 Classifying Windows of Unseen Data

Detecting the roles of meeting participants in unseen data is performed as follows: First the unseen test data is split into windows of the same size as was used during the training regime. Then the speech activity and keywords based features are extracted (using the same keywords as was used during the training) for each participant in each window. Finally these data points are used as input into the trained decision tree, which outputs class membership probabilities for each participant in each window.

5.2.2 Aggregating Evidence to Assign One Role Per Participant

Thus for each participant we get as many probability distributions (over the three roles) as there are windows in the test data. The next step is to aggregate these probabilities over all the windows and arrive at a single role assignment per participant. We employ the simplest possible aggregation method: We compute, for each participant, the average probability of each role over all the windows, and then normalize the three average role probabilities so calculated, so they still sum to 1. In the future we plan to experiment with more sophisticated aggregation mechanisms that jointly optimize the probabilities of the different participants, instead of computing them independently.

At this point, we could assign to each participant his highest probability role. However, we wish to ensure that the set of roles that get assigned to the

participants in a particular meeting are as diverse as possible (since typically meetings are forums at which different people of different expertise convene to exchange information). To ensure such diversity, we apply the following heuristic. Once we have all the average probabilities for all the roles for each participant in a sequence of meetings, we assign roles to participants in *stages*. At each stage we consider all participants not yet assigned roles, and pick that participant–role pair, say (p, r), that has the highest probability value among all pairs under consideration. We assign participant p the role r, and then *discount* (by a constant multiplicative factor) the probability value of all participant–role pairs (p_i, r_j) where p_i is a participant not assigned a role yet, and $r_j = r$. This makes it less likely (but not impossible) that another participant will be assigned this same role r again. This process is repeated until all participants have been assigned a role each.

6 Evaluation

We evaluated the algorithm by computing the accuracy of the detector's role predictions. Specifically, given a meeting sequence we ran the algorithm to assign a role to each meeting participant, and computed the accuracy by calculating the ratio of the number of correct assignments to the total number of participants in the sequence. Note that it is also possible to evaluate the window–by–window classification of the decision tree classifiers; we report results on this evaluation in section 7.1.

To evaluate this participant role detection algorithm, we first trained the algorithm on the training set of meetings. The training phase included keyword list creation, window size optimization, and the actual induction of the decision tree. On the training data, a window size of 300 seconds resulted in the highest accuracy over the training set. The test at the root of the induced tree was whether the participant's rank in terms of speech lengths was 1, in which case he was immediately classified as a *meeting leader*. That is, the tree learnt that the person who spoke the most in a window was most likely the meeting leader. Other tests placed high in the tree included obvious ones such as testing for the keywords *computer* and *printer* to classify a participant as a hardware expert.

We then tested this trained role detector on the testing set of meetings. Recall that the test set had 5 meeting sequences, each consisting of 5 meetings and a total of 20 meeting participants. Over this test set we obtained a role detection accuracy of 83%. A "classifier" that randomly assigns one of the three roles to each participant in a meeting (without regard to the roles assigned to the other participants in the same meeting) would achieve a classification accuracy of 33.3%. Thus, our algorithm significantly beats the random classifier baseline. Note that as mentioned earlier, the experiments in this paper are based on the manually transcribed speech.

7 Further Experiments

7.1 Optimizing the Window Size

As mentioned above, one of the variables to be tuned during the training phase is the size of the window over which to extract speech features. We ran a sequence of experiments to optimize this window size, the results of which are summarized in figure 1. In this set of experiments, we performed the evaluation on two levels of granularity. The larger granularity level was the "meeting sequence" granularity, where we ran the usual evaluation described above. That is, for each participant we first used the classifier to obtain probability distributions over the 3 roles on every window, and then aggregated these distributions to reach a single role assignment for the participant over the entire meeting sequence. This role was compared to the true role of the participant to measure the accuracy of the algorithm. The smaller granularity level was the "window" level, where after obtaining the probability distribution over the three roles for a particular window of a particular participant, we picked the role with the highest probability, and assigned it to the participant *for that window*. Therefore, for each window we had a role assignment that we compared to the true role of the participant, resulting in an accuracy value for the classifier for every window for every participant. Note that the main difference between evaluation at these two granularity levels is that in the "window" granularity, we did not have any aggregation of evidence across multiple windows.

For different window sizes, we plotted the accuracy values obtained on the test set for the two evalu-

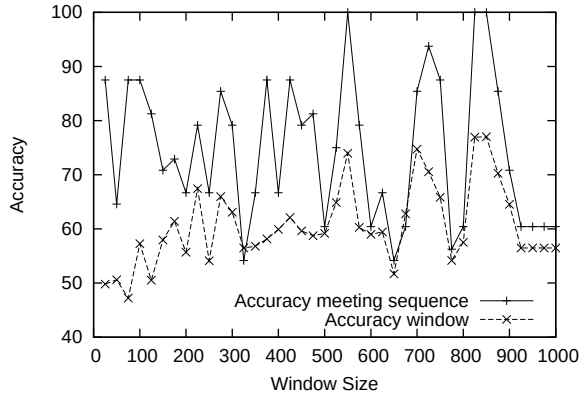


Figure 1: *Effect of Different Window Sizes on Detection Accuracy*

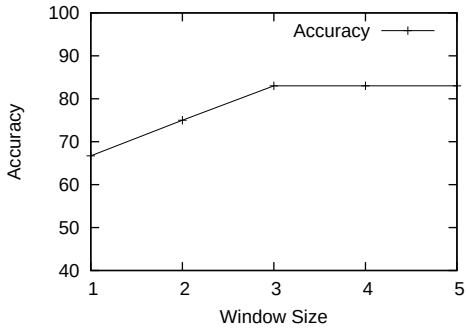


Figure 2: *Accuracy versus Number of Meetings over which Roles were Detected*

ation granularities, as shown in figure 1. Notice that by aggregating the evidence across the windows, the detection accuracy improves for all window sizes. This is to be expected since in the window granularity, the classifier has access to only the information contained in a single window, and is therefore more error prone. However by merging the evidence from many windows, the accuracy improves. As window sizes increase, detection accuracy at the window level improves, because the classifier has more evidence at its disposal to make the decision. However, detection at the meeting sequence level gets steadily worse, potentially because the larger the window size, the fewer the data points it has to aggregate evidence from. These lines will eventually meet when the window size equals the size of the entire meeting sequence.

A valid concern with these results is the high level of noise, particularly in the aggregated detection accuracy over the meeting sequence. One reason for this is that there are far fewer data points at the meeting sequence level than at the window level. With larger data sets (more meeting sequences as well as more participants per meeting) these results may stabilize. Additionally, given the small amount of data, our feature set is quite large, so a more aggressive feature set reduction might help stabilize the results.

7.2 Automatic Improvement over Unseen Data

One of our goals is to create an expertise based role detector system that improves over time as it has access to more and more meetings for a given participant. This is especially important because the roles that a participant plays can change over time; we would like our system to be able to track these changes. In the Y2 Scenario Data that we have used in this current work, the roles do not change from meeting to meeting. However observe that our evidence aggregation algorithm fuses information from all the meetings in a specific sequence of meetings to arrive at a single role assignment for each participant.

To quantify the effect of this aggregation we computed the role detection accuracy using different numbers of meetings from each sequence. Specifically, we computed the accuracy of the role detection over the test data using only the last meeting of each sequence, only the last 2 meetings of each sequence, and so on until we used every meeting in every sequence. The results are summarized in figure 2. When using only the last meeting in the sequence to assign roles to the participants, the accuracy is only 66.7%, when using the last two meetings, the accuracy is 75%, and using the last three, four or all meetings results in an accuracy of 83%. Thus, the accuracy improves as we have more meetings to combine evidence from, as is expected. However the accuracy levels off at 83% when using three or more meetings, perhaps because there is no new information to be gained by adding a fourth or a fifth meeting.

8 Conclusions and Future Work

In this paper we have discussed our current approach to detecting the functional and expertise based roles of meeting participants. We have induced decision

trees that use simple and robust speech based features to perform the role detection. We have used a very simple evidence aggregation mechanism to arrive at a single role assignment per meeting participant over a sequence of meetings, and have shown that we can achieve up to 83% accuracy on unseen test data using this mechanism. Additionally we have shown that by aggregating evidence across a sequence of meetings, we perform better than if we were to use a single meeting to perform the role detection. As future work we plan to remove the constraints that we have currently imposed – namely, we will attempt to learn new roles in test data that do not exist in training data. Additionally, we will attempt to use this role information as inputs to downstream meeting understanding tasks such as automatic topic detection and action item detection.

9 Acknowledgements

This work was supported by DARPA grant NBCH-D-03-0010. The content of the information in this publication does not necessarily reflect the position or the policy of the US Government, and no official endorsement should be inferred.

References

S. Banerjee and A. I. Rudnicky. 2004. Using simple speech-based features to detect the state of a meeting and the roles of the meeting participants. In *Proceedings of the 8th International Conference on Spoken Language Processing (Interspeech 2004 – ICSLP)*, Jeju Island, Korea.

S. Banerjee, J. Cohen, T. Quisel, A. Chan, Y. Patodia, Z. Al-Bawab, R. Zhang, P. Rybski, M. Veloso, A. Black, R. Stern, R. Rosenfeld, and A. I. Rudnicky. 2004. Creating multi-modal, user–centric records of meetings with the Carnegie Mellon meeting recorder architecture. In *Proceedings of the ICASSP Meeting Recognition Workshop*, Montreal, Canada.

S. Banerjee, C. Rose, and A. I. Rudnicky. 2005. The necessity of a meeting recording and playback system, and the benefit of topic–level annotations to meeting browsing. In *Proceedings of the Tenth International Conference on Human-Computer Interaction*, Rome, Italy, September.

CALO. 2003. http://www.ai.sri.com/project/CALO.

M. Galley, K. McKeown, E. Fosler-Lussier, and Hongyan Jing. 2003. Discourse segmentation of multi–party conversation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, volume 1, pages 562 – 569, Sapporo, Japan.

T. Hain, J. Dines, G. Garau, M. Karafiat, D. Moore, V. Wan, R. Ordelman, and S. Renals. 2005. Transcription of conference room meetings: An investigation. In *Proceedings of Interspeech 2005*, Lisbon, Portugal, September.

F. Metze, Q. Jin, C. Fugen, K. Laskowski, Y. Pan, and T. Schultz. 2004. Issues in meeting transcription – the isl meeting transcription system. In *Proceedings of the 8th International Conference on Spoken Language Processing (Interspeech 2004 – ICSLP)*, Jeju Island, Korea.

G. Murray, S. Renals, and J. Carletta. 2005. Extractive summarization of meeting recordings. In *Proceedings of Interspeech 2005*, Lisbon, Portugal, September.

J. Quinlan. 1986. Induction of decision trees. *Machine Learning*, 1:81–106.

Paul E. Rybski and Manuela M. Veloso. 2004. Using sparse visual data to model human activities in meetings. In *Workshop on Modeling Other Agents from Observations, International Joint Conference on Autonomous Agents and Multi-Agent Systems*.

A. Stolcke, C. Wooters, N. Mirghafori, T. Pirinen, I. Bulyko, D. Gelbart, M. Graciarena, S. Otterson, B. Peskin, and M. Ostendorf. 2004. Progress in meeting recognition: The icsi–sri–uw spring 2004 evaluation system. In *NIST RT04 Meeting Recognition Workshop*, Montreal.

I. Witten and E. Frank. 2000. *Data Mining - Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan–Kaufmann, San Francisco, CA.

Y. Yang and J. Pedersen. 1997. A comparative study on feature selection in text categorization. In *Proceedings of the International Conference on Machine Learning*, pages 412–420, Nashville, US. Morgan Kaufmann Publishers.

Shallow Discourse Structure for Action Item Detection

Matthew Purver, Patrick Ehlen, and John Niekrasz
Center for the Study of Language and Information
Stanford University
Stanford, CA 94305
{mpurver,ehlen,niekrasz}@stanford.edu

Abstract

We investigated automatic *action item detection* from transcripts of multi-party meetings. Unlike previous work (Gruenstein et al., 2005), we use a new hierarchical annotation scheme based on the roles utterances play in the action item assignment process, and propose an approach to automatic detection that promises improved classification accuracy while enabling the extraction of useful information for summarization and reporting.

1 Introduction

Action items are specific kinds of decisions common in multi-party meetings, characterized by the concrete assignment of tasks together with certain properties such as an associated timeframe and reponsible party. Our aims are firstly to automatically detect the regions of discourse which establish action items, so their surface form can be used for a targeted report or summary; and secondly, to identify the important properties of the action items (such as the associated tasks and deadlines) that would foster concise and informative semantically-based reporting (for example, adding task specifications to a user's to-do list). We believe both of these aims are facilitated by taking into account the roles different utterances play in the decision-making process – in short, a shallow notion of discourse structure.

2 Background

Related Work Corston-Oliver et al. (2004) attempted to identify action items in e-mails, using classifiers trained on annotations of individual sentences within each e-mail. Sentences were annotated with one of a set of "dialogue" act classes; one class `Task` included any sentence containing items that seemed appropriate to add to an ongoing to-do list. They report good inter-annotator agreement over their general tagging exercise ($\kappa > 0.8$), although individual figures for the `Task` class are not given. They then concentrated on `Task` sentences, establishing a set of predictive features (in which word n-grams emerged as "highly predictive") and achieved reasonable per-sentence classification performance (with f-scores around 0.6).

While there are related tags for dialogue act tagging schema – like DAMSL (Core and Allen, 1997), which includes tags such as `Action-Directive` and `Commit`, and the ICSI MRDA schema (Shriberg et al., 2004) which includes a `commit` tag – these classes are too general to allow identification of action items specifically. One comparable attempt in spoken discourse took a flat approach, annotating utterances as action-item-related or not (Gruenstein et al., 2005) over the ICSI and ISL meeting corpora (Janin et al., 2003; Burger et al., 2002). Their inter-annotator agreement was low ($\kappa = .36$). While this may have been partly due to their methods, it is notable that (Core and Allen, 1997) reported even lower agreement ($\kappa = .15$) on their `Commit` dialogue acts. Morgan et al. (forthcoming) then used these annotations to attempt auto-

Proceedings of the Analyzing Conversations in Text and Speech (ACTS) Workshop at HLT-NAACL 2006, pages 31–34,
New York City, New York, June 2006.

matic classification, but achieved poor performance (with f-scores around 0.3 at best).

Action Items Action items typically embody the transfer of group responsibility to an individual. This need not be the person who actually performs the action (they might delegate the task to a subordinate), but publicly commits to seeing that the action is carried out; we call this person the *owner* of the action item. Because this action is a social action that is coordinated by more than one person, its initiation is reinforced by *agreement* and uptake among the owner and other participants that the action should and will be done. And to distinguish this action from immediate actions that occur during the meeting and from more vague future actions that are still in the planning stage, an action item will be specified as expected to be carried out within a *timeframe* that begins at some point after the meeting and extends no further than the not-too-distant future. So an action item, as a type of social action, often comprises four components: a *task description*, a *timeframe*, an *owner*, and a round of *agreement* among the owner and others. The related discourse tends to reflect this, and we attempt to exploit this fact here.

3 Baseline Experiments

We applied Gruenstein et al. (2005)'s flat annotation schema to transcripts from a sequence of 5 short related meetings with 3 participants recorded as part of the CALO project. Each meeting was simulated in that its participants were given a scenario, but was not scripted. In order to avoid entirely data- or scenario-specific results (and also to provide an acceptable amount of training data), we then added a random selection of 6 ICSI and 1 ISL meetings from Gruenstein et al. (2005)'s annotations. Like (Corston-Oliver et al., 2004) we used support vector machines (Vapnik, 1995) via the classifier *SVM-light* (Joachims, 1999). Their full set of features are not available to us, but we experimented with combinations of words and n-grams and assessed classification performance via a 5-fold validation on each of the CALO meetings. In each case, we trained classifiers on the other 4 meetings in the CALO sequence, plus the fixed ICSI/ISL training selection. Performance (per utterance, on the binary classification problem) is shown in Table 1; overall f-score figures are poor even on these short meetings. These figures were obtained using words (unigrams, after text normalization and stemming) as features – we investigated other discriminative classifier methods, and the use of 2- and 3-grams as features, but no improvements were gained.

Mtg.	Utts	AI Utts.	Precision	Recall	F-Score
1	191	22	0.31	0.50	0.38
2	156	27	0.36	0.33	0.35
3	196	18	0.28	0.55	0.37
4	212	15	0.20	0.60	0.30
5	198	9	0.19	0.67	0.30

Table 1: Baseline Classification Performance

4 Hierarchical Annotations

Two problems are apparent: firstly, accuracy is lower than desired; secondly, identifying utterances related to action items does not allow us to actually identify those action items and extract their properties (deadline, owner etc.). But if the utterances related to these properties form distinct sub-classes which have their own distinct features, treating them separately and combining the results (along the lines of (Klein et al., 2002)) might allow better performance, while also identifying the utterances where each property's value is extracted. Thus, we produced an annotation schema which distinguishes among these four classes. The first three correspond to the discussion and assignment of the individual properties of the action item (`task description`, `timeframe` and `owner`); the final `agreement` class covers utterances which explicitly show that the action item is agreed upon.

Since the `task description` subclass extracts a description of the task, it must include any utterances that specify the action to be performed, including those that provide required antecedents for anaphoric references. The `owner` subclass includes any utterances that explicitly specify the responsible party (e.g. "I'll take care of that", or "John, we'll leave that to you"), but not those whose function might be taken to do so implicitly (such as agreements by the responsible party). The `timeframe` subclass includes any utterances that explicitly refer to when a task may start or when it is expected to be finished; note that this is often not specified with

a date or temporal expression, but rather e.g. "by the end of next week," or "before the trip to Aruba". Finally, the `agreement` subclass includes any utterances in which people agree that the action should and will be done; not only acknowledgements by the owner themselves, but also when other people express their agreement.

A single utterance may be assigned to more than one class: "**John, you** need to do that **by next Monday**" might count as `owner` and `timeframe`. Likewise, there may be more than one utterance of each class for a single action item: John's response "OK, **I'll** do that" would also be classed as `owner` (as well as `agreement`). While we do not require all of these subclasses to be present for a set of utterances to qualify as denoting an action item, we expect any action item to include most of them.

We applied this annotation schema to the same 12 meetings. Initial reliability between two annotators on the single ISL meeting (chosen as it presented a significantly more complex set of action items than others in this set) was encouraging. The best agreement was achieved on `timeframe` utterances ($\kappa = .86$), with `owner` utterances slightly less good (between $\kappa = .77$), and `agreement` and `description` utterances worse but still acceptable ($\kappa = .73$). Further annotation is in progress.

5 Experiments

We trained individual classifiers for each of the utterance sub-classes, and cross-validated as before. For `agreement` utterances, we used a naive n-gram classifier similar to that of (Webb et al., 2005) for dialogue act detection, scoring utterances via a set of most predictive n-grams of length 1–3 and making a classification decision by comparing the maximum score to a threshold (where the n-grams, their scores and the threshold are automatically extracted from the training data). For `owner`, `timeframe` and `task description` utterances, we used SVMs as before, using word unigrams as features (2- and 3-grams gave no improvement – probably due to the small amount of training data). Performance varied greatly by sub-class (see Table 2), with some (e.g. agreement) achieving higher accuracy than the baseline flat classifications, but others being worse. As there is now significantly less training data available to each sub-class than there was for all utterances grouped together in the baseline experiment, worse performance might be expected; yet some sub-classes perform better. The worst performing class is `owner`. Examination of the data shows that `owner` utterances are more likely than other classes to be assigned to more than one category; they may therefore have more feature overlap with other classes, leading to less accurate classification. Use of relevant sub-strings for training (rather than full utterances) may help; as may part-of-speech information – while proper names may be useful features, the name tokens themselves are sparse and may be better substituted with a generic tag.

Class	Precision	Recall	F-Score
description	0.23	0.41	0.29
owner	0.12	0.28	0.17
timeframe	0.19	0.38	0.26
agreement	0.48	0.44	0.40

Table 2: Sub-class Classification Performance

Even with poor performance for some of the subclassifiers, we should still be able to combine them to get a benefit as long as their true positives correlate better than their false positives (intuitively, if they make mistakes in different places). So far we have only conducted an initial naive experiment, in which we combine the individual classifier decisions in a weighted sum over a window (currently set to 5 utterances). If the sum over the window reaches a given threshold, we hypothesize an action item, and take the highest-confidence utterance given by each sub-classifier in that window to provide the corresponding property. As shown in Table 3, this gives reasonable performance on most meetings, although it does badly on meeting 5 (apparently because no explicit agreement takes place, while our manual weights emphasized agreement).[1] Most encouragingly, the correct examples provide some useful "best" sub-class utterances, from which the relevant properties could be extracted.

These results can probably be significantly improved: rather than sum over the binary classification outputs of each classifier, we can use their confidence scores or posterior probabilities, and learn

[1] Accuracy here is currently assessed only over correct detection of an action item in a window, not correct assignment of all sub-classes.

Mtg.	AIs	Correct	False+	False-	F-Score
1	3	2	1	1	0.67
2	4	1	0	3	0.40
3	5	2	1	3	0.50
4	4	4	0	0	1.00
5	3	0	1	3	0.00

Table 3: Combined Classification Performance

the combination weights to give a more robust approach. There is still a long way to go to evaluate this approach over more data, including the accuracy and utility of the resulting sub-class utterance hypotheses.

6 Discussion and Future Work

So accounting for the structure of action items appears essential to detecting them in spoken discourse. Otherwise, classification accuracy is limited. We believe that accuracy can be improved, and the detected utterances can be used to provide the properties of the action item itself. An interesting question is how and whether the structure we use here relates to discourse structure in more general use. If a relation exists, this would shed light on the decision-making process we are attempting to (begin to) model, and might allow us to use other (more plentiful) annotated data.

Our future efforts focus on annotating more meetings to obtain large training and testing sets. We also wish to examine performance when working from speech recognition hypotheses (as opposed to the human transcripts used here), and the best way to incorporate multiple hypotheses (either as n-best lists or word confusion networks). We are actively investigating alternative approaches to sub-classifier combination: better performance (and a more robust and trainable overall system) might be obtained by using a Bayesian network, or a maximum entropy classifier as used by (Klein et al., 2002). Finally, we are developing an interface to a new large-vocabulary version of the Gemini parser (Dowding et al., 1993) which will allow us to use semantic parse information as features in the individual sub-class classifiers, and also to extract entity and event representations from the classified utterances for automatic addition of entries to calendars and to-do lists.

References

S. Burger, V. MacLaren, and H. Yu. 2002. The ISL Meeting Corpus: The impact of meeting type on speech style. In *Proceedings of the 6th International Conference on Spoken Language Processing (ICSLP 2002)*.

M. Core and J. Allen. 1997. Coding dialogues with the DAMSL annotation scheme. In D. Traum, editor, *AAAI Fall Symposium on Communicative Action in Humans and Machines*.

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

J. Dowding, J. M. Gawron, D. Appelt, J. Bear, L. Cherny, R. Moore, and D. Moran. 1993. Gemini: A natural language system for spoken language understanding. In *Proc. 31st Annual Meeting of the Association for Computational Linguistics*.

A. Gruenstein, J. Niekrasz, and M. Purver. 2005. Meeting structure annotation: Data and tools. In *Proceedings of the 6th SIGdial Workshop on Discourse and Dialogue*.

A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters. 2003. The ICSI Meeting Corpus. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2003)*.

T. Joachims. 1999. Making large-scale SVM learning practical. In B. Schölkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods – Support Vector Learning*. MIT Press.

D. Klein, K. Toutanova, H. T. Ilhan, S. D. Kamvar, and C. D. Manning. 2002. Combining heterogeneous classifiers for word-sense disambiguation. In *Proceedings of the ACL Workshop on Word Sense Disambiguation: Recent Successes and Future Directions*.

W. Morgan, S. Gupta, and P.-C. Chang. forthcoming. Automatically detecting action items in audio meeting recordings. Ms., under review.

E. Shriberg, R. Dhillon, S. Bhagat, J. Ang, and H. Carvey. 2004. The ICSI Meeting Recorder Dialog Act Corpus. In *Proceedings of the 5th SIGdial Workshop on Discourse and Dialogue*.

S. Siegel and J. N. J. Castellan. 1988. *Nonparametric Statistics for the Behavioral Sciences*. McGraw-Hill.

V. N. Vapnik. 1995. *The Nature of Statistical Learning Theory*. Springer.

N. Webb, M. Hepple, and Y. Wilks. 2005. Dialogue act classification using intra-utterance features. In *Proc. AAAI Workshop on Spoken Language Understanding*.

Improving "Email Speech Acts" Analysis via N-gram Selection

Vitor R. Carvalho
Language Technologies Institute
Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh PA
vitor@cs.cmu.edu

William W. Cohen
Machine Learning Department
Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh PA
wcohen@cs.cmu.edu

Abstract

In email conversational analysis, it is often useful to trace the the intents behind each message exchange. In this paper, we consider classification of email messages as to whether or not they contain certain intents or email-acts, such as "propose a meeting" or "commit to a task". We demonstrate that exploiting the contextual information in the messages can noticeably improve email-act classification. More specifically, we describe a combination of n-gram sequence features with careful message preprocessing that is highly effective for this task. Compared to a previous study (Cohen et al., 2004), this representation reduces the classification error rates by 26.4% on average. Finally, we introduce Ciranda: a new open source toolkit for email speech act prediction.

1 Introduction

One important use of work-related email is negotiating and delegating shared tasks and subtasks. To provide intelligent email automated assistance, it is desirable to be able to automatically detect the *intent* of an email message—for example, to determine if the email contains a request, a commitment by the sender to perform some task, or an amendment to an earlier proposal. Successfully adding such a semantic layer to email communication is still a challenge to current email clients.

In a previous work, Cohen et al. (2004) used text classification methods to detect "email speech acts". Based on the ideas from Speech Act Theory (Searle, 1975) and guided by analysis of several email corpora, they defined a set of "email acts" (e.g., *Request*, *Deliver*, *Propose*, *Commit*) and then classified emails as containing or not a specific act. Cohen et al. (2004) showed that machine learning algorithms can learn the proposed email-act categories reasonably well. It was also shown that there is an acceptable level of human agreement over the categories.

A method for accurate classification of email into such categories would have many potential applications. For instance, it could be used to help users track the status of ongoing joint activities, improving task delegation and coordination. Email speech acts could also be used to iteratively learn user's tasks in a desktop environment (Khoussainov and Kushmerick, 2005). Email acts classification could also be applied to predict hierarchy positions in structured organizations or email-centered teams (Leusky, 2004); predicting leadership positions can be useful to analyze behavior in teams without an explicitly assigned leader.

By using only single words as features, Cohen et al. (2004) disregarded a very important linguistic aspect of the speech act inference task: the textual context. For instance, the specific sequence of tokens "Can you give me" can be more informative to detect a *Request* act than the words "can", "you", "give" and "me" separately. Similarly, the word sequence "I will call you" may be a much stronger indication of a *Commit* act than the four words separately. More generally, because so many specific

Proceedings of the Analyzing Conversations in Text and Speech (ACTS) Workshop at HLT-NAACL 2006, pages 35–41,
New York City, New York, June 2006.

sequence of words (or n-grams) are inherently associated with the intent of an email message, one would expect that exploiting this linguistic aspect of the messages would improve email-act classification.

In the current work we exploit the linguistic aspects of the problem by a careful combination of n-gram feature extraction and message preprocessing. After preprocessing the messages to detect entities, punctuation, pronouns, dates and times, we generate a new feature set by extracting all possible term sequences with a length of 1, 2, 3, 4 or 5 tokens.

Using this n-gram based representation in classification experiments, we obtained a relative average drop of 26.4% in error rate when compared to the original Cohen et al. (2004) paper. Also, ranking the most "meaningful" n-grams based on Information Gain score (Yang and Pedersen, 1997) revealed an impressive agreement with the linguistic intuition behind the email speech acts.

We finalize this work introducing *Ciranda*: an open source package for Email Speech Act prediction. Among other features, Ciranda provides an easy interface for feature extraction and feature selection, outputs the prediction confidence, and allows retraining using several learning algorithms.

2 "Email-Acts" Taxonomy and Applications

A taxonomy of speech acts applied to email communication (email-acts) is described and motivated in (Cohen et al., 2004). The taxonomy was divided into *verbs* and *nouns*, and each email message is represented by one or more verb-noun pairs. For example, an email proposing a meeting and also requesting a project report would have the labels *Propose-Meeting* and *Request-Data*.

The relevant part of the taxonomy is shown in Figure 1. Very briefly, a *Request* asks the recipient to perform some activity; a *Propose* message proposes a joint activity (i.e., asks the recipient to perform some activity and commits the sender); a *Commit* message commits the sender to some future course of action; *Data* is information, or a pointer to information, delivered to the recipient; and a *Meeting* is a joint activity that is constrained in time and (usually) space.

Several possible verbs/nouns were not considered here (such as *Refuse*, *Greet*, and *Remind*), either because they occurred very infrequently in the corpus, or because they did not appear to be important for task-tracking. The most common verbs found in the labeled datasets were *Deliver*, *Request*, *Commit*, and *Propose*, and the most common nouns were *Meeting* and *deliveredData* (abbreviated as *dData* henceforth).

In our modeling, a single email message may have multiple *verbs-nouns* pairs.

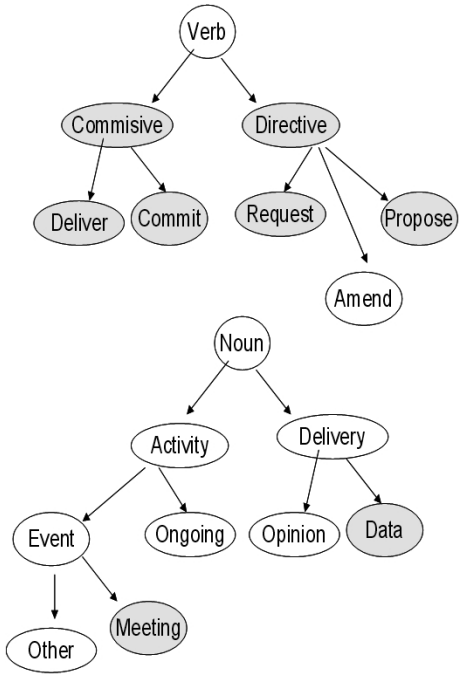


Figure 1: Taxonomy of email-acts used in experiments. Shaded nodes are the ones for which a classifier was constructed.

Cohen et al. (2004) showed that machine learning algorithms can learn the proposed email-act categories reasonably well. It was also shown that there is an acceptable level of human agreement over the categories. In experiments using different human annotators, Kappa values between 0.72 and 0.85 were obtained. The Kappa statistic (Carletta, 1996) is typically used to measure the human inter-rater agreement. Its values ranges from -1 (complete disagreement) to +1 (perfect agreement) and it is defined as (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).

3 The Corpus

The *CSpace* email corpus used in this paper contains approximately 15,000 email messages collected from a management course at Carnegie Mellon University. This corpus originated from working groups who signed agreements to make certain parts of their email accessible to researchers. In this course, 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.,). The email tends to be very task-oriented, with many instances of task delegation and negotiation.

Messages were mostly exchanged with members of the same team. Accordingly, we partitioned the corpus into subsets according to the teams. The 1F3 team dataset has 351 messages total, while the 2F2, 3F2, 4F4 and 11F1 teams have, respectively, 341, 443, 403 and 176 messages. All 1716 messages were labeled according to the taxonomy in Figure 1.

4 N-gram Features

In this section we detail the preprocessing step and the feature selection applied to all email acts.

4.1 Preprocessing

Before extracting the n-grams features, a sequence of preprocessing steps was applied to all email messages in order to emphasize the linguistic aspects of the problem. Unless otherwise mentioned, all preprocessing procedures were applied to all acts.

Initially, forwarded messages quoted inside email messages were deleted. Also, signature files and quoted text from previous messages were removed from all messages using a technique described elsewhere (Carvalho and Cohen, 2004). A similar cleaning procedure was executed by Cohen et al. (2004).

Some types of punctuation marks (“,;:.)(][”) were removed, as were extra spaces and extra page breaks. We then perform some basic substitutions such as: from “*’m*” to “ *am*”, from “*’re*” to “ *are*”, from “*’ll*” to “ *will*”, from “*won’t*” to “*will not*”, from “*doesn’t*” to “*does not*” and from “*’d*” to “ *would*”.

Any sequence of one or more numbers was replaced by the symbol “[number]”. The pattern “[number]:[number]” was replaced with “[hour]”. The expressions “*pm* or *am*” were replaced by “[pm]”. “[wwhh]” denoted the words “*why*, *where*, *who*, *what* or *when*”. The words “*I*, *we*, *you*, *he*, *she* or *they*” were replaced by “[person]”. Days of the week (“*Monday*, *Tuesday*, ..., *Sunday*”) and their short versions (i.e., “*Mon*, *Tue*, *Wed*, ..., *Sun*”) were replaced by “[day]”. The words “*after*, *before* or *during*” were replaced by “[aaafter]”. The pronouns “*me*, *her*, *him*, *us* or *them*” were substituted by “[me]”. The typical filename types “*.doc*, *.xls*, *.txt*, *.pdf*, *.rtf* and *.ppt*” were replaced by “.[filetype]”. A list with some of these substitutions is illustrated in Table 1.

Symbol	Pattern
[number]	any sequence of numbers
[hour]	[number]:[number]
[wwhh]	“why, where, who, what, or when”
[day]	the strings “Monday, Tuesday, ..., or Sunday”
[day]	the strings “Mon, Tue, Wed, ..., or Sun”
[pm]	the strings “P.M., PM, A.M. or AM”
[me]	the pronouns “me, her, him, us or them”
[person]	the pronouns “I, we, you, he, she or they”
[aaafter]	the strings “after, before or during”
[filetype]	the strings “.doc, .pdf, .ppt, .txt, or .xls”

Table 1: Some PreProcessing Substitution Patterns

For the *Commit* act only, references to the first person were removed from the symbol [person] — i.e., [person] was used to replace “he, she or they”. The rationale is that n-grams containing the pronoun “I” are typically among the most meaningful for this act (as shall be detailed in Section 4.2).

4.2 Most Meaningful N-grams

After preprocessing the 1716 email messages, n-gram sequence features were extracted. In this paper, n-gram features are all possible sequences of length 1 (unigrams or 1-gram), 2 (bigram or 2-gram), 3 (trigram or 3-gram), 4 (4-gram) and 5 (5-gram) terms. After extracting all n-grams, the new dataset had more than 347500 different features. It would be interesting to know which of these n-grams are the “most meaningful” for each one of email speech acts.

1-gram	2-gram	3-gram	4-gram	5-gram
?	do [person]	[person] need to	[wwhh] do [person] think	[wwhh] do [person] think ?
please	? [person]	[wwhh] do [person]	do [person] need to	let [me] know [wwhh] [person]
[wwhh]	could [person]	let [me] know	and let [me] know	a call [number]-[number]
could	[person] please	would [person]	call [number]-[number]	give [me] a call [number]
do	? thanks	do [person] think	would be able to	please give give [me] a call
can	are [person]	are [person] meeting	[person] think [person] need	[person] would be able to
of	can [person]	could [person] please	let [me] know [wwhh]	take a look at it
[me]	need to	do [person] need	do [person] think ?	[person] think [person] need to

Table 2: Request Act:Top eight N-grams Selected by Information Gain.

One possible way to accomplish this is using some feature selection method. By computing the Information Gain score (Forman, 2003; Yang and Pedersen, 1997) of each feature, we were able to rank the most "meaningful" n-gram sequence for each speech act. The final rankings are illustrated in Tables 2 and 3.

Table 2 shows the most meaningful n-grams for the *Request* act. The top features clearly agree with the linguistic intuition behind the idea of a *Request* email act. This agreement is present not only in the frequent 1g features, but also in the 2-grams, 3-grams, 4-grams and 5-grams. For instance, sentences such as "What do you think ?" or "let me know what you ..." can be instantiations of the top two 5-grams, and are typically used indicating a request in email communication.

Table 3 illustrates the top fifteen 4-grams for all email speech acts selected by Information Gain. The *Commit* act reflects the general idea of agreeing to do some task, or to participate in some meeting. As we can see, the list with the top 4-grams reflects the intuition of commitment very well. When accepting or committing to a task, it is usual to write emails using "Tomorrow is good for me" or "I will put the document under your door" or "I think I can finish this task by 7" or even "I will try to bring this tomorrow". The list even has some other interesting 4-grams that can be easily associated to very specific commitment situations, such as "I will bring copies" and "I will be there".

Another act in Table 3 that visibly agrees with its linguistic intuition is *Meeting*. The 4-grams listed are usual constructions associated with either negotiating a meeting time/location ("[day] at [hour][pm]"), agreeing to meet ("is good for [me]") or describing the goals of the meeting ("to go over the").

The top features associated with the *dData* act in Table 3 are also closely related to its general intuition. Here the idea is delivering or requesting some data: a table inside the message, an attachment, a document, a report, a link to a file, a url, etc. And indeed, it seems to be exactly the case in Table 3: some of the top 4-grams indicate the presence of an attachment (e.g., "forwarded message begins here"), some features suggest the address or link where a file can be found (e.g., "in my public directory" or "in the etc directory"), some features request an action to access/read the data (e.g., "please take a look") and some features indicate the presence of data inside the email message, possibly formatted as a table (e.g., "[date] [hour] [number] [number]" or "[date] [day] [number] [day]").

From Table 3, the *Propose* act seems closely related to the *Meeting* act. In fact, by checking the labeled dataset, most of the *Proposal*s were associated with *Meeting*s. Some of the features that are not necessarily associated with *Meeting* are " [person] would like to", "please let me know" and "was hoping [person] could".

The *Deliver* email speech act is associated with two large sets of actions: delivery of data and delivery of information in general. Because of this generality, is not straightforward to list the most meaningful n-grams associated with this act. Table 3 shows a variety of features that can be associated with a *Deliver* act. As we shall see in Section 5, the *Deliver* act has the highest error rate in the classification task.

In summary, selecting the top n-gram features via Information Gain revealed an impressive agreement with the linguistic intuition behind the different email speech acts.

Request	Commit	Meeting
[wwhh] do [person] think	is good for [me]	[day] at [hour] [pm]
do [person] need to	is fine with [me]	on [day] at [hour]
and let [me] know	i will see [person]	[person] can meet at
call [number]-[number]	i think i can	[person] meet at [hour]
would be able to	i will put the	will be in the
[person] think [person] need	i will try to	is good for [me]
let [me] know [wwhh]	i will be there	to meet at [hour]
do [person] think ?	will look for [person]	at [hour] in the
[person] need to get	$[number] per person	[person] will see [person]
? [person] need to	am done with the	meet at [hour] in
a copy of our	at [hour] i will	[number] at [hour] [pm]
do [person] have any	[day] is fine with	to go over the
[person] get a chance	each of us will	[person] will be in
[me] know [wwhh]	i will bring copies	let's plan to meet
that would be great	i will do the	meet at [hour] [pm]
dData	**Propose**	**Deliver**
– forwarded message begins	[person] would like to	forwarded message begins here
forwarded message begins here	would like to meet	[number] [number] [number] [number]
is in my public	please let [me] know	is good for [me]
in my public directory	to meet with [person]	if [person] have any
[person] have placed the	[person] meet at [hour]	if fine with me
please take a look	would [person] like to	in my public directory
[day] [hour] [number] [number]	[person] can meet tomorrow	[person] will try to
[number] [day] [number] [hour]	an hour or so	is in my public
[date] [day] [number] [day]	meet at [hour] in	will be able to
in our game directory	like to get together	just wanted to let
in the etc directory	[hour] [pm] in the	[pm] in the lobby
the file name is	[after] [hour] or [after]	[person] will be able
is in our game	[person] will be available	please take a look
fyi – forwarded message	think [person] can meet	can meet in the
just put the file	was hoping [person] could	[day] at [hour] is
my public directory under	do [person] want to	in the commons at

Table 3: Top 4-grams Selected by Information Gain

5 Experiments

Here we describe how the classification experiments on the email speech acts dataset were carried out. Using all n-gram features, we performed 5-fold crossvalidation tests over the 1716 email messages. Linear SVM[1] was used as classifier. Results are illustrated in Figure 2.

Figure 2 shows the test error rate of four different experiments (bars) for all email acts. The first bar denotes the error rate obtained by Cohen et al. (2004) in a 5-fold crossvalidation experiment, also using linear SVM. Their dataset had 1354 email messages, and only 1-gram features were extracted.

The second bar illustrates the error rate obtained using only 1-gram features with additional data. In this case, we used 1716 email messages. The third bar represents the the same as the second bar (1-gram features with 1716 messages), with the difference that the emails went through the preprocessing procedure previously described.

The fourth bar shows the error rate when all 1-gram, 2-gram and 3-gram features are used and the 1716 messages go through the preprocessing procedure. The last bar illustrates the error rate when all n-gram features (i.e., 1g+2g+3g+4g+5g) are used in addition to preprocessing in all 1716 messages.

In all acts, a consistent improvement in 1-gram performance is observed when more data is added, i.e., a drop in error rate from the first to the second bar. Therefore, we can conclude that Cohen et al. (2004) could have obtained better results if they had used more labeled data.

A comparison between the second and third bars reveals the extent to which preprocessing seems to help classification based on 1-grams only. As we can see, no significant performance difference can be observed: for most acts the relative difference is

[1]We used the LIBSVM implementation (Chang and Lin, 2001) with default parameters.

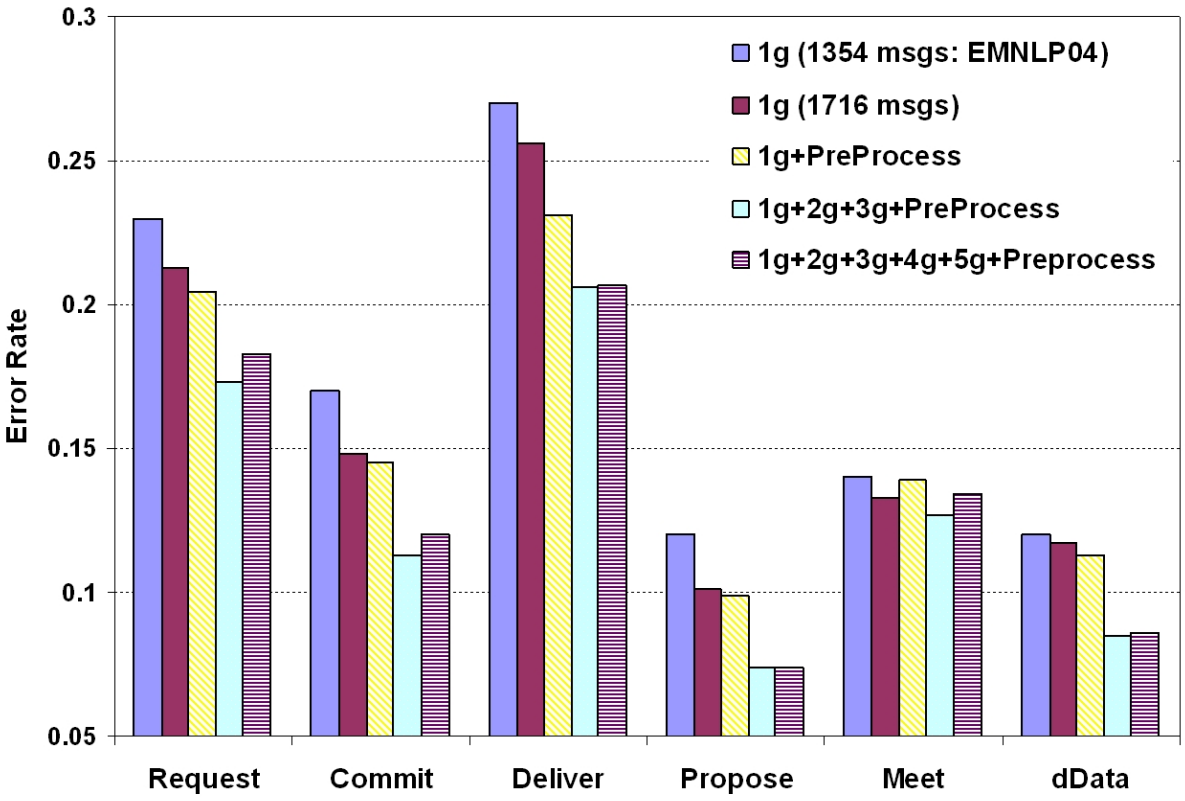


Figure 2: Error Rate 5-fold Crossvalidation Experiment

very small, and in one or maybe two acts some small improvement can be noticed.

A much larger performance improvement can be seen between the fourth and third bars. This reflects the power of the contextual features: using all 1-grams, 2-grams and 3-grams is considerably more powerful than using only 1-gram features. This significant difference can be observed in all acts. Compared to the original values from (Cohen et al., 2004), we observed a relative error rate drop of 24.7% in the *Request* act, 33.3% in the *Commit* act, 23.7% for the *Deliver* act, 38.3% for the *Propose* act, 9.2% for *Meeting* and 29.1% in the *dData* act. In average, a relative improvement of 26.4% in error rate.

We also considered adding the 4-gram and 5-gram features to the best system. As pictured in the last bar of Figure 2, this addition did not seem to improve the performance and, in some cases, even a small increase in error rate was observed. We believe this was caused by the insufficient amount of labeled data in these tests; and the 4-gram and 5-gram features are likely to improve the performance of this system if more labeled data becomes available.

Precision versus recall curves of the *Request* act classification task are illustrated in Figure 3. The curve on the top shows the *Request* act performance when the preprocessing step cues and n-grams proposed in Section 4 are applied. For the bottom curve, only 1g features were used. These two curves correspond to the second bar (bottom curve) and forth bar (top curve) in Figure 2. Figure 3 clearly shows that both recall and precision are improved by using the contextual features.

To summarize, these results confirm the intuition that contextual information (n-grams) can be very effective in the task of email speech act classification.

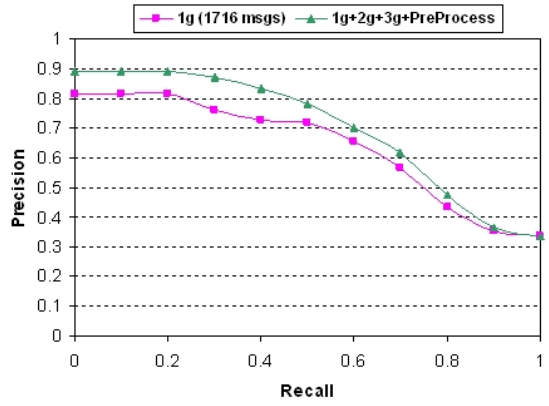


Figure 3: Precision versus Recall of the Request Act Classification

6 The Ciranda Package

Ciranda is an open source package for Email Speech Act prediction built on the top of the Minorthird package (Cohen, 2004). Among other features, Ciranda allows customized feature engineering, extraction and selection. Email Speech Act classifiers can be easily retrained using any learning algorithm from the Minorthird package. Ciranda is currently available from `http://www.cs.cmu.edu/~vitor`.

7 Conclusions

In this work we considered the problem of automatically detecting the intents behind email messages using a shallow semantic taxonomy called "email speech acts" (Cohen et al., 2004). We were interested in the task of classifying whether or not an email message contains acts such as "propose a meeting" or "deliver data".

By exploiting contextual information in emails such as n-gram sequences, we were able to noticeably improve the classification performance on this task. Compared to the original study (Cohen et al., 2004), this representation reduced the classification error rates by 26.4% on average. Improvements of more than 30% were observed for some acts (*Propose* and *Commit*).

We also showed that the selection of the top n-gram features via Information Gain revealed an impressive agreement with the linguistic intuition behind the different email speech acts.

References

[Carletta1996] Jean Carletta. 1996. Assessing agreement on classification tasks: The kappa statistic. *Computational Linguistics*, 22(2):249–254.

[Carvalho and Cohen2004] Vitor R. Carvalho and William W. Cohen. 2004. Learning to extract signature and reply lines from email. In *Proceedings of the Conference on Email and Anti-Spam*, Palo Alto, CA.

[Chang and Lin2001] Chih-Chung Chang and Chih-Jen Lin, 2001. *LIBSVM: a library for support vector machines*. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[Cohen et al.2004] William W. Cohen, Vitor R. Carvalho, and Tom M. Mitchell. 2004. Learning to classify email into "speech acts". In *Proceedings of Empirical Methods in Natural Language Processing*, pages 309–316, Barcelona, Spain, July.

[Cohen2004] William W. Cohen, 2004. *Minorthird: Methods for Identifying Names and Ontological Relations in Text using Heuristics for Inducing Regularities from Data.* `http://minorthird.sourceforge.net`.

[Forman2003] George Forman. 2003. An extensive empirical study of feature selection metrics for text classification. *The Journal of Machine Learning Research*, 3:1289–1305.

[Khoussainov and Kushmerick2005] Rinat Khoussainov and Nicholas Kushmerick. 2005. Email task management: An iterative relational learning approach. In *Conference on Email and Anti-Spam (CEAS'2005)*.

[Kraut et al.] R.E. Kraut, S.R. Fussell, F.J. Lerch, and A. Espinosa. Coordination in teams: Evidence from a simulated management game. To appear in the Journal of Organizational Behavior.

[Leusky2004] Anton Leusky. 2004. Email is a stage: Discovering people roles from email archives. In *ACM Conference on Research and Development in Information Retrieval (SIGIR)*.

[Searle1975] J. R. Searle. 1975. A taxonomy of illocutionary acts. In *In K. Gunderson (Ed.), Language, Mind and Knowledge.*, pages 344–369, Minneapolis, MN. University of Minnesota Press.

[Yang and Pedersen1997] Yiming Yang and Jan O. Pedersen. 1997. A comparative study on feature selection in text categorization. In *Proceedings of ICML-97, 14th International Conference on Machine Learning*, pages 412–420.

Topic Segmentation of Dialogue

Jaime Arguello
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA 15217
jarguell@andrew.cmu.edu

Carolyn Rosé
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA 15217
cprose@cs.cmu.edu

Abstract

We introduce a novel topic segmentation approach that combines evidence of topic shifts from lexical cohesion with linguistic evidence such as syntactically distinct features of segment initial and final contributions. Our evaluation shows that this hybrid approach outperforms state-of-the-art algorithms even when applied to loosely structured, spontaneous dialogue. Further analysis reveals that using dialogue exchanges versus dialogue contributions improves topic segmentation quality.

1 Introduction

In this paper we explore the problem of topic segmentation of dialogue. Use of topic-based models of dialogue has played a role in information retrieval (Oard et al., 2004), information extraction (Baufaden, 2001), and summarization (Zechner, 2001), just to name a few applications. However, most previous work on automatic topic segmentation has focused primarily on segmentation of expository text. This paper presents a survey of the state-of-the-art in topic segmentation technology. Using the definition of topic segment from (Passonneau and Litman, 1993) applied to two different dialogue corpora, we present an evaluation including a detailed error analysis, illustrating why approaches designed for expository text do not generalize well to dialogue.

We first demonstrate a significant advantage of our hybrid, supervised learning approach called Museli, a multi-source evidence integration approach, over competing algorithms. We then extend the basic Museli algorithm by introducing an intermediate level of analysis based on Sinclair and Coulthard's notion of a dialogue exchange (Sinclair and Coulthard, 1975). We show that both our baseline and Museli approaches obtain a significant improvement when using perfect, hand-labeled dialogue exchanges, typically in the order of 2-3 contributions, as the atomic discourse unit in comparison to using the contribution as the unit of analysis. We further evaluate our success towards automatic classification of exchange boundaries using the same Museli framework.

2 Defining Topic

In the most general sense, the challenge of topic segmentation can be construed as the task of finding locations in the discourse where the focus shifts from one topic to another. Thus, it is not possible to address topic segmentation of dialogue without first addressing the question of what a "topic" is. We began with the goal of adopting a definition of topic that meets three criteria. First, it should be reproducible by human annotators. Second, it should not rely heavily on domain-specific knowledge or knowledge of the task structure. Finally, it should be grounded in generally accepted principles of discourse structure.

The last point addresses a subtle, but important, criterion necessary to adequately serve downstream applications using our dialogue segmentation. Topic analysis of dialogue concerns itself mainly with thematic content. However, boundaries should be placed in locations that are natural turning points in the discourse. Shifts in topic should be readily recognizable from surface characteristics of the language.

With these goals in mind, we adopted a definition of "topic" that builds upon Passonneau and Litman's seminal work on segmentation of monologue (Passonneau and Litman, 1993). They found that human annotators can successfully accomplish a flat monologue segmentation using an informal notion of speaker intention.

Proceedings of the Analyzing Conversations in Text and Speech (ACTS) Workshop at HLT-NAACL 2006, pages 42–49,
New York City, New York, June 2006.

Dialogue is inherently hierarchical in structure. However, a flat segmentation model is an adequate approximation. Passonneau and Litman's pilot studies confirmed previously published results (Rotondo, 1984) that human annotators cannot reliably agree on a hierarchical segmentation of monologue. Using a stack-based hierarchical model of discourse, Flammia (1998) found that 90% of all information-bearing dialogue turns referred to the discourse purpose at the top of the stack.

We adopt a flat model of topic segmentation based on discourse segment purpose, where a shift in topic corresponds to a shift in purpose that is acknowledged and acted upon by both conversational participants. We place topic boundaries on contributions that introduce a speaker's intention to shift the purpose of the discourse, while ignoring expressed intentions to shift discourse purposes that are not taken up by the other participant. We adopt the dialogue contribution as the basic unit of analysis, refraining from placing topic boundaries within a contribution. This decision is analogous to Hearst's (Hearst, 1994, 1997) decision to shift the TextTiling induced boundaries to their nearest reference paragraph boundary.

We evaluated the reproducibility of our notion of topic segment boundaries by assessing inter-coder reliability over 10% of the corpus (see Section 5.1). Three annotators were given a 10 page coding manual with explanation of our informal definition of shared discourse segment purpose as well as examples of segmented dialogues. Pairwise inter-coder agreement was above 0.7 for all pairs of annotators.

3 Previous Work

Existing topic segmentation approaches can be loosely classified into two types: (1) lexical cohesion models, and (2) content-oriented models. The underlying assumption in lexical cohesion models is that a shift in term distribution signals a shift in topic (Halliday and Hassan, 1976). The best known algorithm based on this idea is TextTiling (Hearst, 1997). In TextTiling, a sliding window is passed over the vector-space representation of the text. At each position, the cosine correlation between the upper and lower regions of the sliding window is compared with that of the peak cosine correlation values to the left and right of the window. A segment boundary is predicted when the magnitude of the difference exceeds a threshold.

One drawback to relying on term co-occurrence to signal topic continuity is that synonyms or related terms are treated as thematically-unrelated. One proposed solution to this problem is Latent Semantic Analysis (LSA) (Landauer and Dumais, 1997). Two LSA-based algorithms for segmentation are described in (Foltz, 1998) and (Olney and Cai, 2005). Foltz's approach differs from TextTiling mainly in its use of an LSA-based vector space model. Olney and Cai address a problem not addressed by TextTiling or Foltz's approach, which is that cohesion is not just a function of the repetition of thematically-related terms, but also a function of the presentation of new information in reference to information already presented. Their orthonormal basis approach allows for segmentation based on *relevance* and *informativity*.

Content-oriented models, such as (Barzilay and Lee, 2004), rely on the re-occurrence of patterns of topics over multiple realizations of thematically similar discourses, such as a series of newspaper articles about similar events. Their approach utilizes a hidden Markov model where states correspond to topics and state transition probabilities correspond to topic shifts. To obtain the desired number of topics (states), text spans of uniform length (individual contributions, in our case) are clustered. Then, state emission probabilities are induced using smoothed cluster-specific language models. Transition probabilities are induced by considering the proportion of documents in which a contribution assigned to the source cluster (state) immediately precedes a contribution assigned to the target cluster (state). Following an EM-like approach, contributions are reassigned to states until the algorithm converges.

4 Overview of Museli Approach

We cast the segmentation problem as a binary classification problem where each contribution is classified as NEW_TOPIC if it introduces a new topic and SAME_TOPIC otherwise. In our hybrid Museli approach, we combined lexical cohesion with features that have the potential to capture something about the linguistic style that marks shifts in topic. Table 1 lists our features.

Feature	Description
Lexical Cohesion	Cosine correlation of adjacent regions in the discourse. Term vectors of adjacent regions are stemmed and stopwords are removed.
Word-unigram	Unigrams in previous and current contributions
Word-bigram	Bigrams in previous and current contributions
Punctuation	Punctuation of previous and current contributions.
Part-of-Speech (POS) Bigram	POS-Bigrams in previous and current contributions.
Time Difference	Time difference between previous and current contribution, normalized by: (X – MIN)/ (MAX – MIN), where X corresponds to *this* time difference and MIN & MAX are with respect to the whole corpus.
Content Contribution	Binary-valued, is there a non-stopword term in the current contribution?
Contribution Length	Number of words in the current contribution, normalized by: (X – MIN) / (MAX – MIN).
Previous Agent[1]	Binary-valued, was the speaker of the previous contribution the *student* or the *tutor*?

Table 1. Museli Features.

We found that using a Naïve Bayes classifier with an attribute selection wrapper using the chi-square test for ranking attributes performed better than other state-of-the-art machine learning algorithms on our task, perhaps because of the evidence integration oriented nature of the problem. We conducted our evaluation using 10-fold cross-validation, being careful not to include instances from the same dialogue in both the training and test sets on any fold to avoid biasing the trained model with idiosyncratic communicative patterns associated with individual dialogue participants.

To capitalize on differences in conversational behavior between participants assigned to different roles in the conversation (i.e., student and tutor), we learn separate models for each role. This decision is motivated by observations that participants with different speaker-roles, each with different goals in the conversation, introduce topics with a different frequency, introduce different types of topics, and may introduce topics in a different style that displays their status in the conversation. For instance, a tutor may be more likely to introduce new topics with a contribution that ends with an *imperative.* A student may be more likely to introduce new topics with a contribution that ends with a *wh-question*. Dissimilar agent-roles also occur in other domains such as Travel Agent and Customer in flight booking scenarios.

Using the complete set of features enumerated above, we perform feature selection on the training data for each fold of the cross-validation separately, training a model with the top 1000 features, and applying that trained model to the test data. Examples of high ranking features output by our chi-squared feature selection wrapper confirm our intuition that initial and final contributions of a segment are marked differently. Moreover, the highest ranked features are different for our two speaker-roles. Some features highly-correlated with student-initiated segments are *am_trying*, *should*, *what_is*, and *PUNCT_question*, which relate to student questions and requests for information. Some features highly-correlated with tutor-initiated segments include *ok_lets*, *do*, *see_what*, and *BEGIN_VERB* (the POS of the first word in the contribution is VERB), which characterize imperatives, and features such as *now*, *next*, and *first*, which characterize instructional task ordering.

5 Evaluation

We evaluate Museli in comparison to the best performing state-of-the-art approaches, demonstrating that our hybrid Museli approach outperforms all of these approaches on two different dialogue corpora by a statistically significant margin ($p < .01$), in one case reducing the probability of error, as measured by P_k (Beeferman et al., 1999), to about 10%.

5.1 Experimental Corpora

We used two different dialogue corpora from the educational domain for our evaluation. Both corpora constitute of dialogues between a student and

[1] The current contribution's agent is implicit in the fact that we learn separate models for each agent-role (student & tutor).

a tutor (speakers with asymmetric roles) and both were collected via chat software. The first corpus, which we call the *Olney & Cai corpus*, is a set of dialogues selected randomly from the same corpus Olney and Cai obtained their corpus from (Olney and Cai, 2005). The dialogues discuss problems related to Newton's Three Laws of Motion. The second corpus, the *Thermo corpus*, is a locally collected corpus of thermodynamics tutoring dialogues, in which tutor-student pairs work together to solve an optimization task. Table 2 shows corpus statistics from both corpora.

	Olney & Cai Corpus	**Thermo Corpus**
#Dialogues	42	22
Conts./Dialogue	195.40	217.90
Conts./Topic	24.00	13.31
Topics/Dialogue	8.14	16.36
Words/Cont.	28.63	5.12
Student Conts.	4113	1431
Tutor Conts.	4094	3363

Table 2. Evaluation Corpora Statistics

Both corpora seem adequate for attempting to harness systematic differences in how speakers with asymmetric roles may initiate or close topic segments. The Thermo corpus is particularly appropriate for addressing the research question of how to automatically segment natural, *spontaneous* dialogue. The exploratory task is more loosely structured than many task-oriented domains investigated in the dialogue community, such as flight reservation or meeting scheduling. Students can interrupt with questions and tutors can digress in any way they feel may benefit the completion of the task. In the Olney and Cai corpus, the same 10 physics problems are addressed in each session and the interaction is almost exclusively a tutor initiation followed by student response, evident from the nearly equal number of student and tutor contributions.

5.2 Baseline Approaches

We evaluate Museli against the following four algorithms: (1) Olney and Cai (Ortho), (2) Barzilay and Lee (B&L), (3) TextTiling (TT), and (4) Foltz.

As opposed to the other baseline algorithms, (Olney and Cai, 2005) applied their orthonormal basis approach specifically to dialogue, and prior to this work, report the highest numbers for topic segmentation of dialogue. Barzilay and Lee's approach is the state of the art in modeling topic shifts in monologue text. Our application of B&L to dialogue attempts to harness any existing and recognizable redundancy in topic-flow across our dialogues for the purpose of topic segmentation.

We chose TextTiling for its seminal contribution to monologue segmentation. TextTiling and Foltz consider lexical cohesion as their only evidence of topic shifts. Applying these approaches to dialogue segmentation sheds light on how term distribution in dialogue differs from that of expository monologue text (e.g. news articles). The Foltz and Ortho approaches require a trained LSA space, which we prepared the same way as described in (Olney and Cai, 2005). Any parameter tuning for approaches other than our Museli was computed over the entire test set, giving baseline algorithms the maximum advantage.

In addition to these approaches, we include segmentation results from three degenerate approaches: (1) classifying *all* contributions as NEW_TOPIC (ALL), (2) classifying *no* contributions as NEW_TOPIC (NONE), and (3) classifying contributions as NEW_TOPIC at *uniform intervals* (EVEN), separated by the average reference topic length (see Table 2).

As a means for comparison, we adopt two evaluation metrics: P_k and f-measure. An extensive argument in support of P_k's robustness (if k is set to ½ the average reference topic length) is presented in (Beeferman, et al. 1999). P_k measures the probability of misclassifying two contributions a distance of k contributions apart, where the classification question is *are the two contributions part of the same topic segment or not?* P_k is the likelihood of misclassifying two contributions, thus lower P_k values are preferred over higher ones. It equally captures the effect of false-negatives and false-positives and favors predictions that that are closer to the reference boundaries. F-measure punishes false positives equally, regardless of their distance to reference boundaries.

5.3 Results

Table 3 shows our evaluation results. Note that lower values of P_k are preferred over higher ones. The opposite is true of F-measure. In both corpora, the Museli approach performed significantly better than all other approaches ($p < .01$).

	Olney and Cai Corpus		Thermo Corpus	
	P_k	F	P_k	F
NONE	0.4897	--	0.4900	--
ALL	0.5180	--	0.5100	--
EVEN	0.5117	--	0.5131	--
TT	0.6240	0.1475	0.5353	0.1614
B&L	0.6351	0.1747	0.5086	0.1512
Foltz	0.3270	0.3492	0.5058	0.1180
Ortho	0.2754	0.6012	0.4898	0.2111
Museli	**0.1051**	**0.8013**	**0.4043**	**0.3693**

Table 3. Results on both corpora

5.4 Error Analysis

Results for all approaches are better on the Olney and Cai corpus than the Thermo corpus. The Thermo corpus differs profoundly from the Olney and Cai corpus in ways that very likely influenced the performance. For instance, in the Thermo corpus each dialogue contribution is on average 5 words long, whereas in the Olney and Cai corpus each dialogue contribution contains an average of 28 words. Thus, the vector space representation of the dialogue contributions is more sparse in the Thermo corpus, which makes shifts in lexical coherence less reliable as topic shift indicators.

In terms of P_k, TextTiling (TT) performed worse than the degenerate algorithms. TextTiling measures the term overlap between adjacent regions in the discourse. However, dialogue contributions are often terse or even contentless. This produces many islands of contribution-sequences for which the local lexical coherence is zero. TextTiling wrongly classifies all of these as starts of new topics. A heuristic improvement to prevent TextTiling from placing topic boundaries at every point along a sequence of contributions failed to produce a statistically significant improvement.

The Foltz and the Ortho approaches rely on LSA to provide strategic semantic generalizations capable of detecting shifts in topic. Following (Olney and Cai, 2005), we built our LSA space using dialogue contributions as the atomic text unit. In corpora such as the Thermo corpus, however, this may not be effective due to the brevity of contributions.

Barzilay and Lee's algorithm (B&L) did not generalize well to either dialogue corpus. One reason could be that probabilistic methods, such as their approach, require that reference topics have significantly different language models, which was not true in either of our evaluation corpora. We also noticed a number of instances in the dialogue corpora where participants referred to information from previous topic segments, which consequently may have blurred the distinction between the language models assigned to different topics.

6 Dialogue Exchanges

Although results are reliably better than our baseline algorithms in both corpora, there is much room for improvement, especially in the more spontaneous Thermo corpus. We believe that an improvement can come from a multi-layer segmentation approach, where a first pass segments a dialogue into dialogue exchanges and a second classifier assigns topic shifts based on *exchange initial contributions*. Dialogue is hierarchical in nature. Topic and topic shift comprise only one of the many lenses through which dialogue behaves in seemingly structured ways. Thus, it seems logical that exploiting more fine-grained sub-parts of dialogue than our definition of topic might help us do better at predicting shifts in topic. One such subpart of dialogue is the notion of dialogue exchange, typically between 2-3 contributions.

Stubbs (1983) motivates the definition of an exchange with the following observation. In theory, there is no limit to the number of possible responses to the clause "*Is Harry at home?*". However, constraints are imposed on the interpretation of the contribution that follows it: *yes* or *no*. Such a constraint is central to the concept of a dialogue exchange. Informally, an exchange is made from an initiation, for which the possibilities are open-ended, followed by dialogue contributions that are pre-classified and thus increasingly restricted. A contribution is part of the next exchange when the constraint on its communicative act is lifted.

Sinclair and Coulthard (1975) introduce a more formal definition of exchange with their Initiative-Response-Feedback or IRF structure. An initiation produces a response and a response happens as direct consequence to an initiation. Feedback serves to close an exchange. Sinclair and Coulthard posit that if exchanges constitute the minimal unit of interaction, IRF is a primary structure of interactive discourse in general.

To measure the benefits of exchange boundaries in detecting topic shift in dialogue, we coded the Thermo corpus with exchanges following Sinclair

and Coulthard's IRF structure. The coder who labeled dialogue exchanges had no knowledge of our definition of topic or our intention to do topic-analyses of the corpus. Any correlation between exchange boundaries and topic boundaries is not a bias introduced during the hand-labeling process.

7 Topic Segmentation with Exchanges

In our corpus, as we believe is true in domain-general dialogue, knowledge of an exchange-boundary increases the probability of a topic-boundary significantly. One way to quantify this relation is with the following observation. In our experimental Thermo corpus, there are 4794 dialogue contributions, 360 topic shifts, and 1074 exchange shifts. Using maximum likelihood estimation, the likelihood of being correct if we say that a randomly chosen contribution is a topic shift is 0.075 (# topic shifts / # contributions). However, the likelihood of being correct if we have prior knowledge that an exchange-shift also occurs in that contribution is 0.25. Thus, knowledge that the contribution introduces a new exchange increases our confidence that it also introduces a new topic. More importantly, the probability that a contribution does not mark a topic shift, given that it does not mark an exchange-shift, is 0.98. Thus, exchanges show great promise in narrowing the search-space of tentative topic shifts.

In addition to possibly narrowing the space of tentative topic-boundaries, exchanges are helpful in that they provide more coarse-grain building blocks for segmentation algorithms that rely on term-distribution as a proxy for dialogue coherence, such as TextTiling (Hearst, 1994, 1997), the Foltz algorithm (Foltz, 1998), Orthonormal Basis (Olney and Cai, 2005), and Barzilay and Lee's content modeling approach (Barzilay and Lee, 2004). At the heart of all these approaches is the assumption that a change in term distribution signals a shift in topic. When applied to dialogue, the major weakness of these approaches is that contributions are often times contentless: terse and absent of thematically meaningful terms. Thus, a more coarse-grained discourse unit is needed.

8 Barzilay and Lee with Exchanges

Barzilay and Lee (2004) offer an attractive frame work for constructing a context-specific Hidden Markov Model (HMM) of topic drift. In our initial evaluation, we used dialogue contributions as the atomic discourse unit. Using contributions, our application of Barzilay and Lee's algorithm for segmenting dialogue fails at least in part because the model learns states that are not thematically meaningful, but instead relate to other systematic phenomena in dialogue, such as fixed expressions and discourse cues. Figure 1 shows the cluster (state) size distribution in terms of the percentage of the total discourse units (exchanges vs. contributions) in the Thermo corpus assigned to each cluster. In the horizontal axis, clusters (states) are sorted by size from largest to smallest.

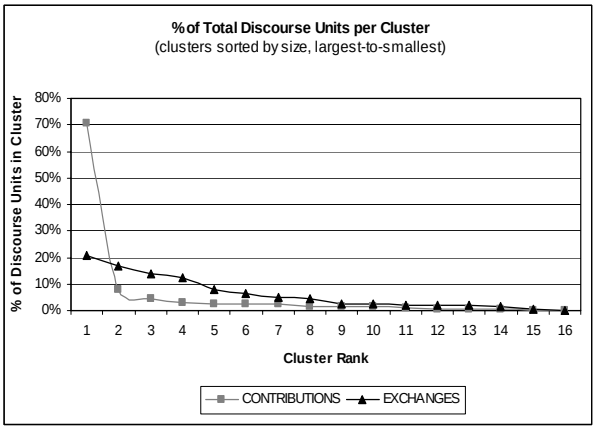


Figure 1. Exchanges produce a more evenly distributed cluster size distribution.

The largest cluster contains 70% of all contributions in the corpus. The second largest cluster only generates 10% of the contributions. In contrast, when using exchanges as the atomic unit, the cluster size distribution is less skewed and corresponds more closely to a topic analysis performed by a domain expert. In this analysis, the number of desired cluster (states), which is an input to the algorithm, was set to 16, the same number identified in a domain expert's analysis of the Thermo corpus. Examples of such topics include high-level ones such as *greeting*, *setup initialization*, and *general thermo concepts*, as well as task-specific ones like *sensitivity analysis* and *regeneration*.

A closer examination of the clusters (states) confirms our intuition that systematic topic-independent phenomena in dialogue, coupled with the terse nature of contributions in spontaneous dialogue, leads to an overly skewed cluster size distribution. Examining the terms with the highest emission probabilities, the largest states contain

topical terms like *cycle*, *efficiency*, *increase*, *quality*, *plot*, and *turbine* intermixed with terms like *think*, *you*, *right*, *make*, *yeah*, *fine*, and *ok*. Also the sets of topical terms in these larger states do not seem coherent with respect to the expert induced topics. This suggests that thematically ambiguous fixed expressions blur the distinction between the different topic-centered language models, producing an overly heavy-tailed cluster size distribution.

One might argue that a possible solution to this problem would be to remove these fixed expressions as part of pre-processing. However, that requires knowledge of the particular domain and knowledge of the interaction style characteristic to the context. We believe that a more robust solution is to use exchanges as the atomic unit of discourse.

9 Evaluation with Exchanges

To show the value of dialogue exchanges in topic segmentation, in this section we re-formulate our problem from classifying contributions into NEW_TOPIC and SAME_TOPIC to classifying exchange initial contributions into NEW_TOPIC and SAME_TOPIC. For all algorithms, we consider only predictions that coincide with hand-coded exchange initial contributions. We show that, except for our own Museli approach, using exchange boundaries improves segmentation quality across *all* algorithms ($p < .05$) when compared to their respective counterparts that ignore exchanges. Using exchanges gives the Museli approach a significant advantage based on F-measure ($p < .05$), but only a marginally significant advantage based on P_k. These results confirm our intuition that what gives our Museli approach an advantage over baseline algorithms is its ability to harness the lexical, syntactic, and phrasal cues that mark shifts in topic. Given that shift-in-topic correlates highly with shift-in-exchange, these features are discriminatory in both respects.

Of the degenerate strategies in section 5.2, only ALL lends itself to our reformulation of the topic segmentation problem. For the ALL heuristic, we classify *all* exchange initial contributions into NEW_TOPIC. This degenerate heuristic alone produces better results than all algorithms classifying utterances (Table 4). In our implementation of TextTiling (TT) with exchanges, we only consider predictions on contributions that coincide with exchange initial contributions, while ignoring predictions made on contributions that do not introduce a new exchange. Consistent with our evaluation methodology from Section 5, we optimized the window size using the entire corpus and found an optimal window size of 13 contributions. Without exchanges, the optimal window size was 6 contributions. The higher optimal window-size hints to the possibility that by using exchange initial contributions an approach based on lexical cohesion may broaden its horizon without losing precision.

	Thermo Corpus (Contributions)		**Thermo Corpus (Exchanges)**	
	P_k	**F**	**P_k**	**F**
NONE	0.4900	--	N/A	--
ALL	0.5100	--	0.4398	0.3809
EVEN	0.5132	--	N/A	--
TT	0.5353	0.1614	0.4328	0.3031
B&L	0.5086	0.1512	0.3817	0.3840
Foltz	0.5058	0.1180	0.4242	0.3296
Ortho	0.4898	0.2111	0.4398	0.3813
Museli	**0.4043**	**0.3693**	**0.3737**	**0.3897**

Table 4. Results using perfect exchange boundaries

In this version of B&L, we use exchanges to build the initial clusters (states) and the final HMM. B&L with exchanges significantly improves over B&L with contributions, in terms of both P_k and F-measure ($p < .005$) and significantly improves over our ALL heuristic (where all exchange initial contributions introduce a new topic) in terms of P_k ($p < .0005$). Thus, its use of exchanges goes beyond merely narrowing the space of possible NEW_TOPIC contributions: it also uses these more coarse-grained discourse units to build a more thematically-motivated topic model.

Foltz's and Olney and Cai's (Ortho) approach both use an LSA space trained on the dialogue corpus. Instead of training the LSA space with individual contributions, we train the LSA space using exchanges. We hope that by training the space with more contentful text units LSA might capture more topically-meaningful semantic relations. In addition, only exchange initial contributions where used for the logistic regression training phase. Thus, we aim to learn the regression equation that best discriminates between exchange initial contributions that introduce a topic and those that do not. Both Foltz and Ortho improve over their non exchange counterparts, but neither improves over the ALL heuristic by a significant margin.

For Museli with exchanges, we tried both training the model using only exchange initial contributions, and applying our previous model to only exchange initial contributions. Training our models using only exchange initial contributions produced slightly worse results. We believe that the reduction of the amount of training data prevents our models from learning good generalizations. Thus, we trained our models using contributions (as in Section 5) and consider predictions only on exchange initial contributions. The Museli approach offers a significant advantage over TT in terms of P_k and F-measure. Using perfect-exchanges, it is not significantly better than Barzilay and Lee. It is significantly better than Foltz's approach based on F-measure and significantly better than Olney and Cai based on P_k ($p < .05$).

These experiments used hand coded exchange boundaries. We also evaluated our ability to automatically predict exchange boundaries. On the Thermo corpus, Museli was able to predict exchange boundaries with precision = 0.48, recall = 0.62, f-measure = 0.53, and $P_k = 0.14$.

10 Conclusions and Current Directions

In this paper we addressed the problem of automatic topic segmentation of spontaneous dialogue. We demonstrated with an empirical evaluation that state-of-the-art approaches fail on spontaneous dialogue because term distribution alone fails to provide adequate evidence of topic shifts in dialogue.

We have presented a supervised learning algorithm for topic segmentation of dialogue called Museli that combines linguistic features signaling a contribution's function with local context indicators. Our evaluation on two distinct corpora shows a significant improvement over the state-of-the-art algorithms. We have also demonstrated that a significant improvement in performance of state-of-the-art approaches to topic segmentation can be achieved when dialogue exchanges, rather than contributions, are used as the basic unit of discourse. We demonstrated promising results in automatically identifying exchange boundaries.

Acknowledgments

This work was funded by Office of Naval Research, Cognitive and Neural Science Division; grant number N00014-05-1-0043.

References

Regina Barzilay and Lillian Lee. 2004. Catching the drift: Probabilistic Content Models, with Applications to Generation and Summarization. In *Proceedings of HLT-NAACL*, 113 - 120.

Doug Beeferman, Adam Berger, John D. Lafferty. 1999. Statistical Models for Text Segmentation. *Machine Learning*, 34 (1-3): 177-210.

Narjès Boufaden, Guy Lapalme, Yoshua Bengio. 2001. Topic Segmentation: A first stage to Dialog-based Information Extraction. In *Proceedings of NLPRS*.

Giovanni Flammia. 1998. *Discourse Segmentation of Spoken Dialogue, PhD Thesis*. Massachusetts Institute of Technology.

Peter Foltz, Walter Kintsch, and Thomas Landauer. 1998. The measurement of textual cohesion with LSA. *Discourse Processes*, 25, 285-307.

Michael Halliday and Ruqaiya Hasan. 1976. *Cohesion in English*. London: Longman.

Marti Hearst. 1997. TextTiling: Segmenting Text into Multi-Paragragh Subtopic Passages. *Computational Linguistics*, 23(1), 33 – 64.

Thomas Landauer and Susan Dumais. A Solution to Plato's Problem: The Latent Semantic Analysis of Acquisition, Induction, and Representation of Knowledge. *Psychological Review*, 104, 221-240.

Douglas Oard, Bhuvana Ramabhadran, and Samuel Gustman. 2004. Building an Information Retrieval Test Collection for Spontaneous Conversational Speech. In *Proceedings of SIGIR.*

Andrew Olney and Zhiqiang Cai. 2005. An Orthonormal Basis for Topic Segmentation of Tutorial Dialogue. In *Proceedings of HLT/EMNLP*. 971-978.

Rebecca Passonneau and Diane Litman. 1993. Intention-Based Segmentation: Human Reliability and Correlation with Linguistic Cues. In *Proceedings of ACL*, 148 – 155.

John Rotondo, 1984, *Clustering Analysis of Subject Partitions of Text.* Discourse Processes, 7:69-88

John Sinclair and Malcolm Coulthard. 1975. *Towards an Analysis of Discourse: the English Used by Teachers and Pupils*. Oxford University Press.

Michael Stubbs. 1983. *Discourse Analysis. A Sociolinguistic Analysis of Natural Language*. Basil Blackwell.

Klaus Zechner. 2001. *Automatic Summarization of Spoken Dialogues in Unrestricted Domains*. Ph.D. Thesis. Carnegie Mellon University.

ChAT: A Time-Linked System for Conversational Analysis

Michelle L. Gregory **Douglas Love** **Stuart Rose** **Anne Schur**

Pacific Northwest National Laboratory
609 Battelle Blvd
Richland, WA 99354

{michelle.gregory;douglas.love;stuart.rose;anne.schur}@pnl.gov

Abstract

We present a system for analyzing conversational data. The system includes state-of-the-art natural language processing components that have been modified to accommodate the unique nature of conversational data. In addition, we leverage the added richness of conversational data by analyzing various aspects of the participants and their relationships to each other. Our tool provides users with the ability to easily identify topics or persons of interest, including who talked to whom, when, entities that were discussed, etc. Using this tool, one can also isolate more complex networks of information: individuals who may have discussed the same topics but never talked to each other. The tool includes a UI that plots information over time, and a semantic graph that highlights relationships of interest.

1 Introduction

The ability to extract and summarize content from data is a fundamental goal of computational linguistics. As such, a number of tools exist to automatically categorize, cluster, and extract information from documents. However, these tools do not transfer well to data sources that are more conversational in nature, such as multi-party meetings, telephone conversations, email, chat rooms, etc. Given the plethora of these data sources, there is a need to be able to quickly and accurately extract and process pertinent information from these sources without having to cull them manually.

Much of the work on computational analysis of dialogue has focused on automatic topic segmentation of conversational data, and in particular, using features of the discourse to aid in segmentation (Galley et al, 2003; Stolcke et al., 1999; Hirschberg & Hakatani, 1996.). Detailed discourse and conversational analytics have been the focus of much linguistic research and have been used by the computational community for creating models of dialogue to aid in natural language understanding and generation (Allen & Core, 1997; Carletta et al., 1997; van Deemter et al., 2005; Walker et al., 1996). However, there has been much less focus on computational tools that can aid in either the analysis of conversations themselves, or in rendering conversational data in ways such that it can be used with traditional data mining techniques that have been successful for document understanding.

This current work is most similar to the NITE XML Toolkit (Carletta & Kilgour, 2005) which was designed for annotating conversational data. NITE XML is system in which transcripts of conversations are viewable and time aligned with their audio transcripts. It is especially useful for adding annotations to multi-modal data formats. NITE XML is not analysis tool, however. Annotations are generally manually added. In this paper, we present a Conversational Analysis Tool (ChAT) which integrates several language processing tools (topic segmentation, affect scoring, named entity extraction) that can be used to automatically annotate conversational data. The processing components have been specially adapted to deal with conversational data.

ChAT is not an annotation tool, however, it is analysis tool. It includes a UI that combines a variety of data sources onto one screen that enables users to progressively explore conversational data. For example, one can explore who was present in a

Proceedings of the Analyzing Conversations in Text and Speech (ACTS) Workshop at HLT-NAACL 2006, pages 50–57,
New York City, New York, June 2006.

given conversation, what they talked about, and the emotional content of the data. The data can be viewed by time slice or in a semantic graph. The language processing components in ChAT are versatile in that they were developed in modular, open designs so that they can be used independently or be integrated into other analytics tools. We present ChAT architecture and processing components in Section 2. In section 3 we present the UI , with a discussion following in section 4.

2 ChAT Architecture

ChAT is a text processing tool designed to aid in the analysis of any kind of threaded dialogue, including meeting transcripts, telephone transcripts, usenet groups, chat room, email or blogs. Figure 1 illustrates the data processing flow in ChAT.

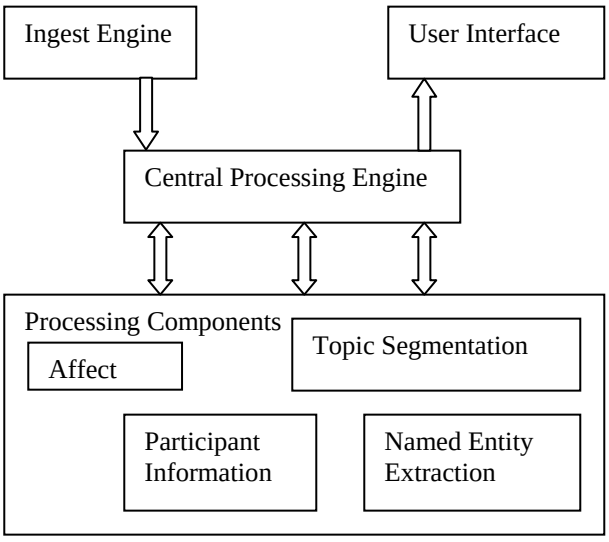


Figure 1: ChAT Architecture.

Data is ingested via an ingest engine, then the central processing engine normalizes the format (time stamp, speaker ID, utterance; one utterance per line). Processing components are called by the central processing engine which provides the input to each component, and collects the output to send to the UI.

We designed the system to be general enough to handle multiple data types. Thus, with the exception of the ingest engine, the processing components are domain and source independent. For example, we did not want the topic segmentation to rely on features specific to a dataset, such as acoustic information from transcripts. Additionally, all processing components have been built as independent plug-ins to the processing engine: The input of one does not rely on the output of the others. This allows for a great deal of flexibility in that a user can choose to include or exclude various processes to suit their needs, or even exchange the components with new tools. We describe each of the processing components in the next section.

2.1 Ingest Engine

The ingest engine is designed to input multiple data sources and transform them into a uniform structure which includes one utterance per line, including time stamp and participant information. So far, we have ingested three data sources. The ICSI meeting corpus (Janin *et al.,* 2003) is a corpus of text transcripts of research meetings. There are 75 meetings in the corpus, lasting 30 minutes to 1.5 hours in duration, with 5-8 participants in each meeting. A subset of these meetings were hand coded for topic segments (Galley, *et al*., 2003). We also used telephone transcripts from the August 14, 2003 power grid failure that resulted in a regional blackout[1]. These data consist of files containing transcripts of multiple telephone conversations between multiple parties. Lastly, we employed a chat room dataset that was built in-house by summer interns who were instructed to play a murder mystery game over chat. Participants took on a character persona as their login and content was based on a predefined scenario, but all interactions were unscripted beyond that.

[1] http://energycommerce.house.gov/108/Hearings/09032003hearing1061/hearing.htm

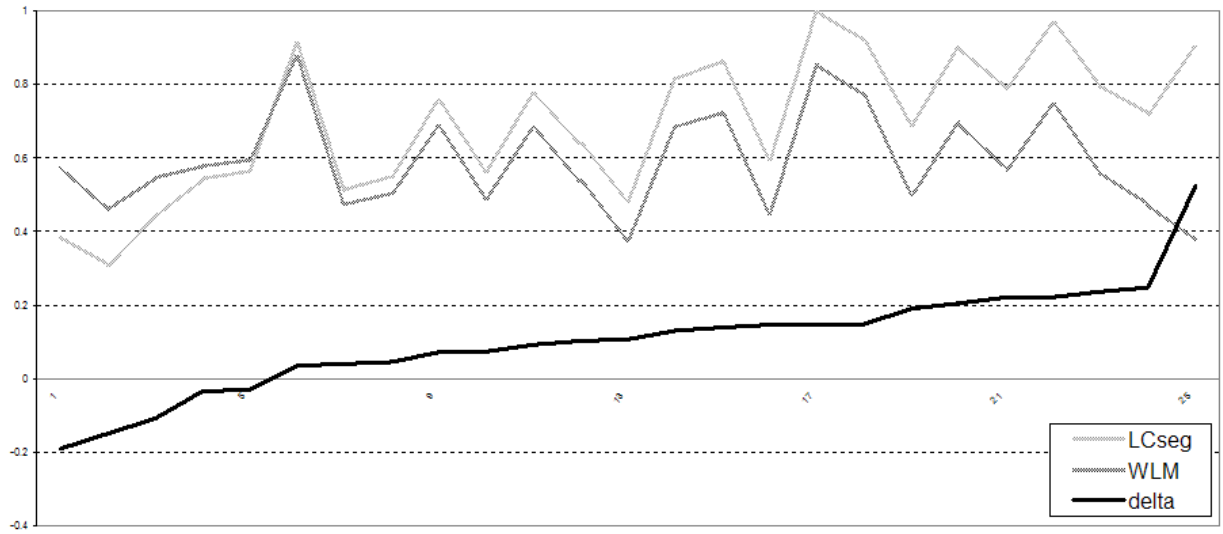


Figure 2. Plot of WindowDiff evaluation metric for LCseg and WLM on meeting corpus. p-value = 0.032121 for two-sample equal variance t-test.

2.2 Topic Segmentation

The output of the ingest process is a list of utterance that include a time (or sequence) stamp, a participant name, and an utterance. Topic segmentation is then performed on the utterances to chunk them into topically cohesive units. Traditionally, algorithms for segmentation have relied on textual cues (Hearst, 1997; Miller et al. 1998; Beeferman et al, 1999; Choi, 2000). These techniques have proved useful in segmenting single authored documents that are rich in content and where there is a great deal of topic continuity. Topic segmentation of conversational data is much more difficult due to often sparse content, intertwined topics, and lack of topic continuity.

Topic segmentation algorithms generally rely on a lexical cohesion signal that requires smoothing in order to eliminate noise from changes of word choices in adjoining statements that do not indicate topic shifts (Hearst, 1997; Barzilay and Elhadad, 1997). Many state of the art techniques use a sliding window for smoothing (Hearst, 1997; Miller et al. 1998; Galley et al., 2003). We employ a windowless method (WLM) for calculating a suitable cohesion signal which does not rely on a sliding window to achieve the requisite smoothing for an effective segmentation. Instead, WLM employs a constrained minimal-spanning tree (MST) algorithm to find and join pairs of elements in a sequence. In most applications, the nearest-neighbor search used by an MST involves an exhaustive, O(N2), search throughout all pairs of elements. However since WLM only requires information on the distance between adjoining elements in the sequence the search space for finding the two closest adjoining elements is linear, O(N), where N is the number of elements in the sequence. We can therefore take advantage of the hierarchical summary structure that an MST algorithm affords while not incurring the performance penalty.

Of particular interest for our research was the success of WLM on threaded dialogue. We evaluated WLM's performance on the ICSI meeting corpus (Janin et al, 2003) by comparing our segmentation results to the results obtained by implementing LCSeg (Galley et al., 2003). Using the 25 hand segmented meetings, our algorithm achieved a significantly better segmentation for 20 out of 25 documents. Figure 2 shows the hypothesized segments from the two algorithms on the ICSI Meeting Corpus.

Topic segmentation of conversational data can be aided by employing features of the discourse or speech environment, such as acoustic cues, etc. (Stolcke et al., 1999; Galley et al., 2003). In this work, we have avoided using data dependent (the

integration of acoustic cues for speech transcripts, for example) features to aid in segmentation because we wanted our system to be as versatile as possible. This approach provides the best segmentation possible for a variety of data sources, regardless of data type.

2.3 Named Entity Extraction

In addition to topics, ChAT also has integrated software to extract the named entities. We use Cicero Lite from the Language Computer Corporation (LCC) for our entity detection (for a product description and evaluation, see Harabagiu et al., 2003). Using a combination of semantic representations and statistical approaches, Cicero Lite isolates approximately 80 entity types. ChAT currently makes use of only a handful of these categories, but can easily be modified to include more. Because named entity extraction relies on cross-utterance dependencies, the main processing engine sends all utterance from a conversation at once rather than an utterance at a time.

2.4 Sentiment Analysis

In addition to topic and entity extraction, conversations can also be analyzed by who participated in them and their relationship to one another and their attitude toward topics they discuss. In an initial attempt to capture participant attitude, we have included a sentiment analysis, or affect, component. Sentiment analysis is conducted via a lexical approach. The lexicon we employed is the General Inquirer (GI) lexicon developed for content analyses of textual data (Stone, 1977). It includes an extensive lexicon of over 11,000 hand coded word stems, and 182 categories, but our implementation is limited to positive (POS) and negative (NEG) axes. In ChAT, every utterance is scored for the number of positive and negative words it contains. We make use of this data by keeping track of the affect of topics in general, as well as the general mood of the participants.

2.5 Participant Roles

Analyzing conversations consists of more than analyzing the topics within them. Inherent to the nature of conversational data are the participants. Using textual cues, one can gain insight into the relationships of participants to each other and the topics. In ChAT we have integrated several simple metrics as indicators of social dynamics amongst the participants. Using simple speaker statistics, such as number of utterances, number of words, etc., we can gain insight to the level of engagement of participants in the conversation. Features we use include:

- The number of utterance
- Proportion of questions versus statements
- Proportion of “unsolicited” statements (ones not preceded by a question mark)

Additionally, we use the same lexical resources as we use for sentiment analysis for indications of personality type. We make use of the lexical categories of strong, weak, power cooperative, and power conflict as indicators of participant roles in the conversational setting. Thus far, we have not conducted any formal evaluation on the sentiment analysis with this data, but our initial studies of our pos and neg categories show a 73% agreement with hand tagged positive and negative segments on a different data set.

3 User Interface

As described in Section 2 on ChAT architecture, the processing components are independent of the UI, but we do have a built-in UI that incorporates the processing components that is designed to aid in analyzing conversations.

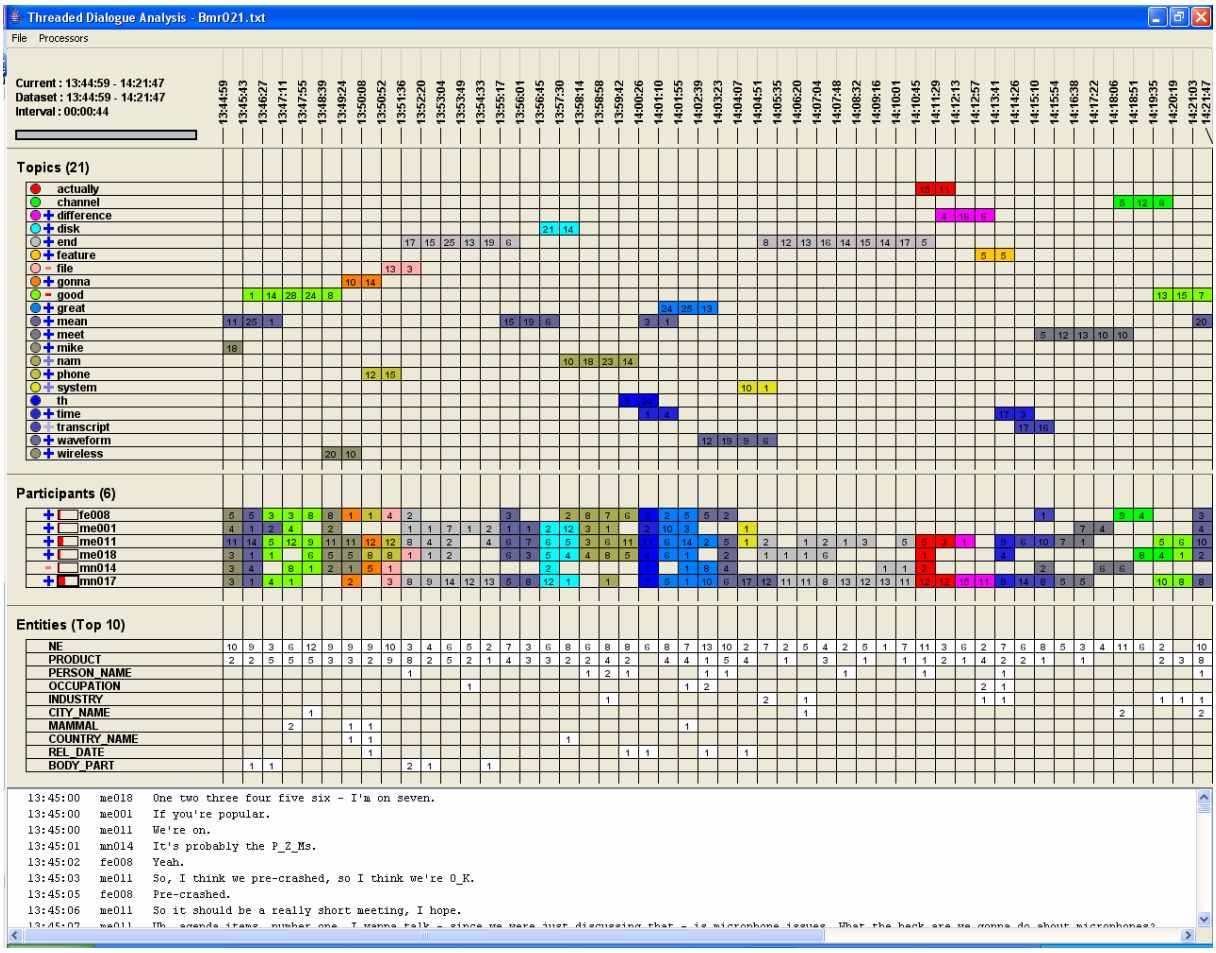


Figure 3. Screen shot of the main UI for ChAT

The components of the system are all linked through the X-axis, representing time, as seen in Figure 3. Depending on the dataset, positions along the time axis are based on either the time stamp or sequential position of the utterance. The default time range is the whole conversation or chat room session, but a narrower range can be selected by dragging in the interval panel at the top of the UI. Note that all of the values for each of the components are recalculated based on the selected time interval. Figure 4 shows that a time selection results in a finer grained subset of the data, allowing one to drill down to specific topics of interest.

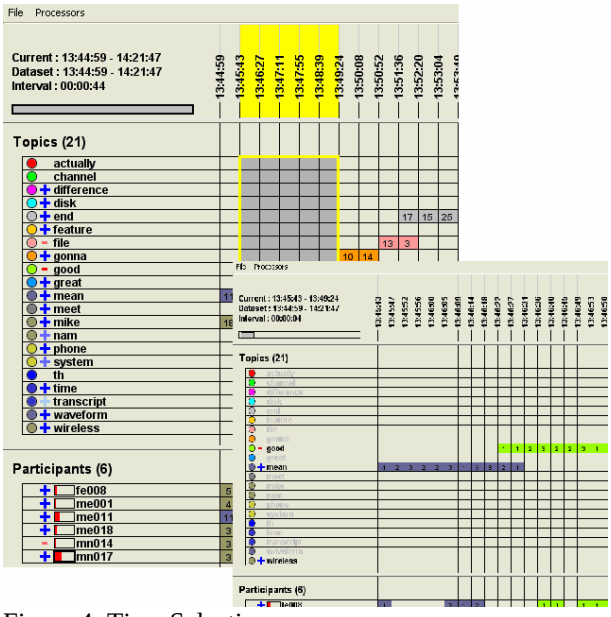


Figure 4: Time Selection.

The number of utterance for a given time frame is indicated by the number inside the box corresponding to the time frame. The number is recalculated as different time frames are selected.

3.1.1 Topics

The central organizing unit in the UI is topics. The topic panel, detailed in Figure 5, consists of three parts: the color key, affect scores, and topic labels. Once a data file is imported into the UI, topic segmentation is performed on the dataset according to the processes outline in Section 3.2. Topic labels are assigned to each topic chunk. Currently, we use the most prevalent word tokens as the label, and the user can control the number of words per label. Each topic segment is assigned a color, which is indicated by the color key. The persistence of a color throughout the time axis indicates which topic is being discussed at any given time. This allows a user to quickly see the distribution of topics of a meeting, for example. It also allows a user to quickly see the participants who discussed a given topic.

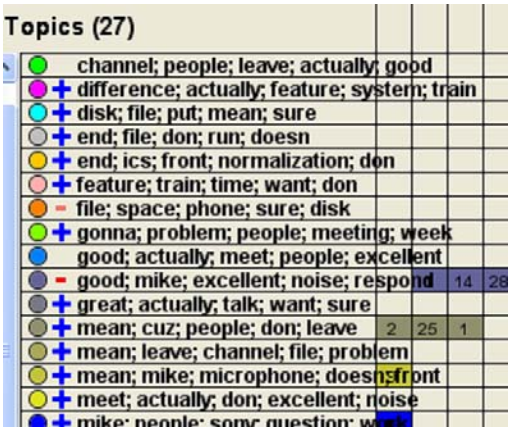


Figure 5. Topic Labels in the Topic Panel.

3.1.2 Affect

Affect scores are computed for each topic by counting the number of POS or NEG affect words in each utterance that comprises a topic within the selected time interval. Affect is measured by the proportion of POS to NEG words in the selected time frame. If the proportion is greater than 0, the score is POS (represented by a +), if it is less than 0, it is NEG (-). The degree of sentiment is indicated by varying shades of color on the + or – symbol.

Note that affect is computed for both topics and participants. An affect score on the topic panel indicates overall affect contained in the utterances present in a given time frame, whereas the affect score in the participant panel indicates overall affect in a given participant's utterances for that time frame.

3.1.3 Participants

The participant panel (Figure 6) consists of three parts: speaker labels, speaker contribution bar, and affect score. The speaker label is displayed in alphabetical order and is grayed out if there are no utterances containing the topic in the selected time frame. The speaker contribution bar, displayed as a horizontal histogram, shows the speaker's proportion of utterances during the time frame. Non question utterances are displayed in red while utterances containing questions are displayed in green as seen. For example, in Figure 6, we can see that speaker me011 did most of the talking (and was generally negative), but speaker me018 had a higher proportion of questions.

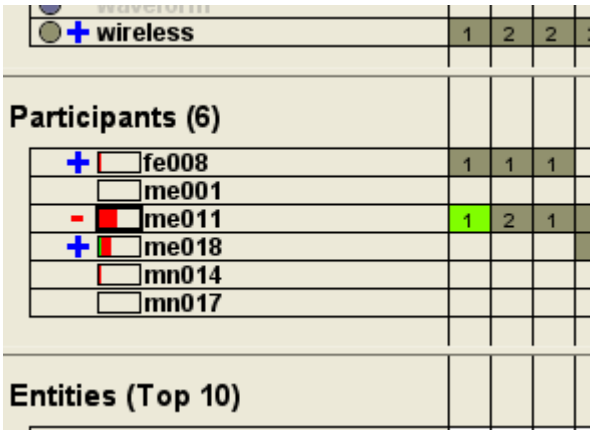


Figure 6. Participant Panel.

3.1.4 Named Entities

In the current implementation, the named entity panel consists of only list of entity labels present in a given time frame. We do not list each named entity because of space constraints, but plan to integrate a scroll bar so that we can display individual entities as opposed to the category labels.

3.2 Semantic Graph

Data that is viewed in the main UI can be sent to a semantic graph for further analysis. The graph allows a user to choose to highlight the relationships associated with a topic, participant, or individual named entity. The user selects objects of interest from a list (see Figure 7), then the graph function organizes a graph according to the chosen object, see Figure 8, that extracts the information from the time-linked view and represent it in a more abstract view that denotes relationships via links and nodes.

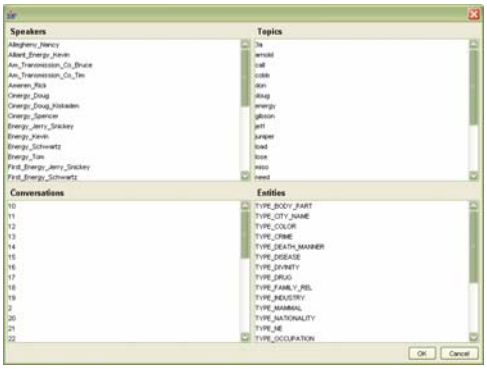


Figure 7. Semantic Graph Node Selection.

The semantic graph can help highlight relationships that might be hard to view in the main UI. For example, Figure 8 represents a subset of the Blackout data in which three participants, indicated by blue, all talked about the same named entity, indicated by green, but never talked to each other, indicated by the red conversation nodes.

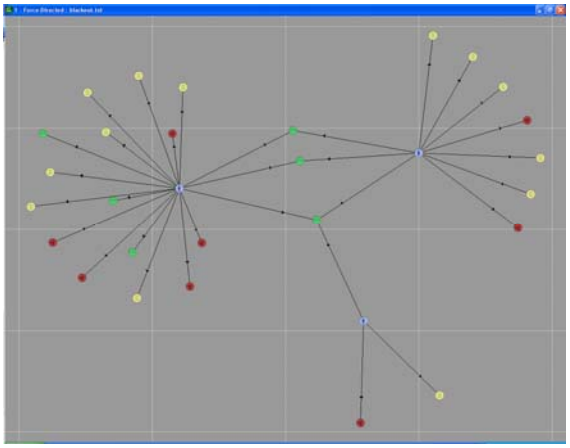


Figure 8. Graph of the Relationship between Three Participants.

4 Discussion

In this paper, we have presented ChAT, a system designed to aid in the analysis of any kind of threaded dialogue. Our system is designed to be flexible in that the UI and processing components work with multiple data types. The processing components can be used independently, or within the UI. The UI aids users in in-depth analysis of individual conversations. The components can be run independent of the UI and in batch, resulting in an xml document containing the original transcripts and the metadata added by the processing components. This functionality allows the data to be manipulated by traditional text mining techniques, or to be viewed in any other visualization.

We have not performed user evaluation on the interface. Our topic segmentation performs better than the current state of the art, and named-entity extraction we have integrated is commercial grade. We are currently working on an evaluation of the affect scoring. While our topic segmentation is good, we are working to improve the labels we use for the topics. Most importantly, we plan on addressing the usefulness of the UI with user studies.

References

R. Barzilay and M. Elhadad. Using lexical chains for text summarization. In *Proc.of the Intelligent Scalable Text Summarization Workshop (ISTS'97), ACL*, 1997.

D. Beeferman, A. Berger, and J. Lafferty. 1999. Statisticalmodels for text segmentation. *Machine Learning*, 34(1–3):177–210.

Carletta, J.C. and Kilgour, J. (2005) The NITE XML Toolkit Meets the ICSI Meeting Corpus: Import, Annotation, and Browsing. MLMI'04: *Proceedings of the Workshop on Machine Learning for Multimodal Interaction.* Samy Bengio and Herve Bourlard, eds. Springer-Verlag Lecture Notes in Computer Science Volume 3361.

F. Choi. 2000. Advances in domain independent linear text segmentation. In *Proc. of NAACL'00*.

van Deemter, Emiel Krahmer & Mariët Theune. 2005 .Real versus template-based Natural Language Generation: a false opposition? *Computational Linguistics* 31(1), pages 15-24.

M. Galley, Kathleen McKeown, Eric Fosler-Lussier, Hongyan Jing. Discourse Segmentation of Multi-

party Conversation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL-03)*. July 2003. Sapporo, Japan.

S. Harabagiu, D. Moldovan, C. Clark, M. Bowden, J. Williams, and J. Bensley. 2003. Answer Mining by Combining Extraction Techniques with Abductive Reasoning, *Proceedings of the Twelfth Text Retrieval Conference* (*TREC*):375.

M. A. Hearst. TexTiling: Segmenting text info multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64, 1997.

J. Hirschberg and C. Nakatani. A prosodic analysis of discourse segments in direction-giving monologues. In *Proc. ACL*, pp. 286–293, Santa Cruz, CA, 1996.

A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart,N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters. 2003. The ICSI meeting corpus. In *Proc. of ICASSP-03, Hong Kong*

N. E. Miller, P. Wong, M. Brewster, and H. Foote. TOPIC ISLANDS - A wavelet-based text visualization system. In David Ebert, Hans Hagen, and Holly Rushmeier, editors, IEEE Visualization '98, pages 189-196, 1998.

A. Stolcke, E. Shriberg, D. Hakkani-Tur, G. Tur, Z. Rivlin, K. Sonmez (1999), Combining Words and Speech Prosody for Automatic Topic Segmentation. *Proc. DARPA Broadcast News Workshop*, pp. 61-64, Herndon, VA.

P. Stone, 1977. Thematic text analysis: new agendas for analyzing text content. in *Text Analysis for the Social Sciences* ed. Carl Roberts. Lawrence Erlbaum Associates.

Pragmatic Discourse Representation Theory

Yafa Al-Raheb
National Centre for Language Technology
Dublin City University
Ireland
yafa.alraheb@gmail.com

Abstract

This paper presents a pragmatic approach to Discourse Representation Theory (DRT) in an attempt to address the pragmatic limitations of DRT (Werth 1999; Simons 2003). To achieve a more pragmatic DRT model, this paper extends standard DRT framework to incorporate more pragmatic elements such as representing agents' cognitive states and the complex process through which agents recognize utterances employing the linguistic content in forming mental representations of other agent's cognitive states. The paper gives focus to the usually ignored link in DRT literature between speaker beliefs and the linguistic content, and between the linguistic content and hearer's beliefs.

1 Introduction

Developments in dynamic semantics, resulting in DRT, have led to a framework suitable for the representation of linguistic phenomena (van Eijck and Kamp 1997). This is specifically due to the fact that, recognizing the importance of context, DRT concentrates on updating the context with the processing of each utterance. In addition, DRT can also be viewed as an agent's mental model of the world and not just a representation of the discourse. It is for these reasons that DRT holds great potential for incorporating more pragmatic phenomena.

However, despite the suitability of DRT for representing linguistic phenomena, some pragmatic limitations have been noted in the literature. Simons (2003) remarks that DRT is a theory of semantics and not pragmatics. Werth remarks that 'there is no place in [DRT] for participant roles, setting, background knowledge, purposes, even inferences' (Werth 1999: 65). In general terms, we can say that the pragmatic dimension supplements semantic content by using context and cognitive states of agents in dialogue. The discipline of pragmatics is, therefore, concerned with the process by which agents infer information about elements of another agents' cognitive state such as their beliefs and intentions. Thus, this paper focuses on extending standard DRT pragmatically to model agents' cognitive states in the pragmatic context of dialogue.

2 A More Pragmatic DRT

This section presents a more pragmatic DRT focusing on the relationship between speaker generation and the linguistic content, and between the linguistic content and hearer recognition. Figure 1 represents the link between our representation of the speaker's cognitive state, the speaker's linguistic content and the hearer's cognitive state or DRS (Discourse Representation Structure). This relationship has not to our knowledge been explored in the literature and deserves investigation.

Generally speaking, to generate an utterance, there would be some discrepancy between the speaker's beliefs and the speaker's beliefs about the hearer's beliefs. The discrepancy leads to an utterance, i.e. linguistic content. The linguistic content is the window the hearer has onto the speaker's state of mind. It is what influences hearer *recognition*. By analysis of the linguistic content provided by the speaker, the hearer can propose a hypothesis regarding the speaker's state of mind.

Proceedings of the Analyzing Conversations in Text and Speech (ACTS) Workshop at HLT-NAACL 2006, pages 58–61,
New York City, New York, June 2006.

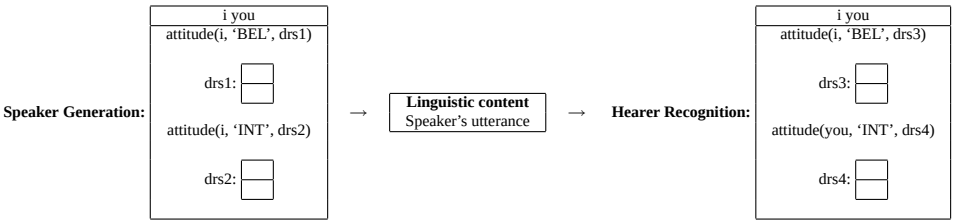


Figure 1: Speaker DRS, Linguistic Content and Hearer DRS

2.1 New DR-Structures

The DRT representation introduced here extends standard DRT language and structure resulting in a suitable pragmatic-based framework for representing this pragmatic link. Separate DRSs are created to represent each agent. DRSs get updated with each new utterance. Each DRS representing an agent's cognitive state includes the two personal reference markers 'i' and 'you'. When 'i' is used in a DRS, it refers to the agent's self within that DRS; i.e. if the agent is the speaker, then 'i' refers to the speaker in the entire DRS. To refer to the other agent, 'you' is used. To follow from the speaker's example, 'you' in this case refers to the hearer. To account for agents' cognitive states and their meta-beliefs, a sub-DRS representing the agent's cognitive state called the *belief DRS* is created to include the speaker's beliefs about the hearer's beliefs. Additionally, a new DRS for representing weaker beliefs called *acceptance* is introduced. The same level of embedding offered to belief DRSs is introduced in acceptance DRSs. Acceptance DRS includes the speaker's acceptance DRS as well as what the speaker takes the hearer to accept. Provided the speaker has sufficient information, the speaker can also have the embedded DRS within the acceptance DRS that represents what the hearer takes the speaker to accept.

In addition to expanding the belief DRS, each agent's cognitive state contains an *intention DRS*. Intention in the sense used here refers to the agent's goals in making an utterance, which are represented by the corresponding dialogue act marked in the intention DRS. The hearer's intention DRS represents the recognized utterance and contains elements of utterance-making generally associated with pragmatics such as the function of an utterance, its dialogue act. This pragmatic enriching strengthens the link between an agent's intentions and the linguistic form uttered. What is proposed is that the intention DRS be designed to include the linguistic content provided within utterances.

To further enhance the link between agents' cognitive states and the linguistic content of their utterances, the intention DRS contains the rich pragmatic information offered by explicitly marking the presupposition (given information) and the assertion (new information) of the current utterance. The intention DRS is a separate DRS from the belief DRS. The beliefs of an agent give the motivation for making an utterance, and the intention DRS represents the speaker's intended message. The recognition of an utterance gives the hearer an insight into the agent's beliefs. Depending upon the particular dialogue represented, the intention DRS could have the speaker's intention, the hearer's intentions or both. The intention DRS functions as the *immediate context*, the one containing the utterance being generated or recognized. The belief and acceptance DRSs function as *background context* containing information pertaining to the dialogue and not just the current utterance. This division of labour context-wise is useful in that the information represented in the intention DRS directly feeds into the speaker's utterance, and is then inferred by the hearer through the linguistic content. The hearer's intention DRS includes the inferred speaker intentions in uttering the current utterance. This gives the flexibility of being able to model information that the hearer has inferred but has not yet decided to accept or believe and is, therefore, not yet included in either the belief or acceptance DRS. For instance, while the hearer in example (1) has recognized S1's utterance, he has not yet accepted S1's utterance. This motivates separating the representation of beliefs from intentions.

(1) S1: Bob's trophy wife is cheating on him.
H1: When did Bob get married?

2.2 Extending DRT Language

In addition to the three DRSs introduced above, in order to make the link between speaker generation, linguistic content, and hearer recognition more explicit, *labels*, 'label$_n$', $_n$ an integer, are introduced. The labels mark the distinction between presupposition and assertion, and the distinction between weak and strong beliefs. Furthermore, the labels can be used to refer to a particular predicate by another complex predicate. The labels increase the expressive power from an essentially first-order formalism to a higher-order formalism. Presuppositions are marked by a presupposition label 'p$_n$'. Similarly, DRSs inside the main speaker or hearer DRS are labeled 'drs$_n$'. Assertions are marked by 'a$_n$' to strengthen the connections between the linguistic form (in the separation between presupposition and assertion) and the representation of beliefs. Believed information labeled 'b$_n$' inside a belief DRS or accepted information labeled 'c$_n$' inside an acceptance DRS can be either presupposed or asserted inside the intention DRS. Thus, the labels in the intention DRS can only be 'p' or 'a'.

Conditions referring to attitudes (acceptance, beliefs, and intentions) have been added to the extended semantics of DRT. Figure 2 shows three embedded DRSs, acceptance DRS, drs2, belief DRS, drs4, and intention DRS, drs6 representing:

(2) A: Tom is buying Mary a puppy.
B: That's sweet.

DRSs are referred to by the attitude describing them. For example, attitude(i,'BEL', drs4) refers to the DRS containing the speaker's beliefs, using the label for the belief DRS, drs4. Other conditions are allowed to employ 'i' as an argument. Attitude(i,'accept', drs2) refers to the DRS containing the speaker's acceptance DRS, using the label for the acceptance DRS, drs2. Attitude(i,'INT', drs6) refers to the DRS containing the speaker's intention in uttering example (2), using the label for the intention DRS, drs6. The speaker's acceptance DRS contains an embedded DRS for the hearer's acceptance DRS, drs2. In this case, it is empty, as no weakly believed propositions have been introduced yet. Similarly, the belief DRS contains space for the speaker's beliefs about the hearer's beliefs, drs5. The intention DRS contains the linguistic content of the utterance that the speaker is about to make, drs6, as well as the relevant dialogue acts.

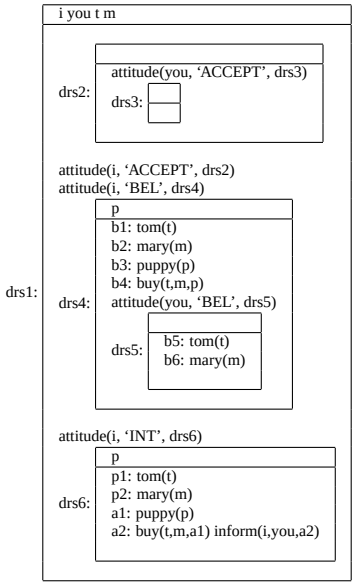


Figure 2: A's initial Cognitive State

In Figure 2, there are essentially three levels of *embedding* in a main DRS. If we look at the belief DRS, the first embedded DRS is the agent's own belief DRS. Level two is the agent's beliefs about the other agent's beliefs DRS. Level three is inserted when necessary and represents the agent's beliefs about the other agent's beliefs about the agent's beliefs DRS. DRSs of the same level of embedding have similar status. For example, the agent's acceptance and belief DRSs have equal status. However, the only discourse referents in common are the ones in the main DRS's universe. Each equal-level embedding has its own set of discourse referents, as well as its own conditions.

Discourse referents of same and higher levels of embedding are accessible to lower levels of embedding and are therefore not represented in the lower level embedding universe. This does not entail that when a lower level embedding makes use of a discourse referent introduced in a higher level embedding the agent and other agent share the same internal or external anchors. For example, when talking about a rabbit, the speaker's representation of rabbit

will be: b1:rabbit(x), whereas the speaker's representation of the hearer's beliefs will be b2:rabbit(x). This is to replace Kamp and Reyle's (1993) use of different discourse referents, where a new discourse referent is used every time the same object or individual is referred to in a new sentence (e.g. rabbit(x), then rabbit(y)). The aim is to avoid having to overtly use the x=y rule every time the same rabbit is referred to. The principles behind the equation predicate are still in place; i.e. every time rabbit is referred to, it is bound to the rabbit already in the context. However, we bind it to the previous properties of rabbit already in context through attaching it to the same discourse referent, rabbit(x).

Both Kamp and Reyle's and our representation face revision when it transpires that the agents in dialogue have different referents in mind. For example, both the speaker and hearer might be talking about 'rabbit'. However, they might have a different 'rabbit' in mind, and assume the other participant is thinking of the rabbit they have in mind. The speaker might have a grey rabbit in mind, whereas the hearer has a white rabbit in mind. In this case, Kamp and Reyle's revision would consist of deleting x=y predicate, and any previous equation predicate that may have been introduced each time rabbit was referred to. In our representation, the revision takes place by changing the other agent's discourse referent, b2:rabbit(x) becomes label2:rabbit(y).

Furthermore, the previous pragmatic extensions to standard DRT have been implemented computationally to approximate a computational model of communication and to enable us to see whether the extended DRT works logically. The implementation relates the linguistic content of utterances to the beliefs and intentions of the agents. The implementation operates with a specific dialogue, which can be modified, within a restricted domain. It seems reasonable to conclude on the basis of the implementation that the conceptual and formal proposals made provide a basis for further development.

3 Conclusion and Further Extensions

This paper pushes the treatment of linguistic phenomena in DRT more towards pragmatics, by bringing more pragmatic elements to the semantic/pragmatic interface which is DRT. It has been the aim of this paper to achieve this by (a) expanding DRT structure to incorporate the pragmatic extensions introduced in this paper, (b) representing the complex process of speakers recognizing utterances and using the linguistic information in forming mental representations of hearers' mental representations, (c) enhancing the link between speaker beliefs, and between the linguistic content and the linguistic content and hearer's beliefs and (d) putting all these extensions and enhancements to the pragmatic side of DRT in a computational model.

While the work presented in this paper offers a more pragmatic approach to DRT, there is still more work to be done on making DRT more pragmatic. The possibility of extending the present treatment to include more agents remains for future work. In addition, future work can employ the intention DRS introduced in this paper, in order to enhance the complexity of the pragmatic representation of speaker/hearer intentions. For instance, embedding turn-taking acts within the intention DRS and relating them to agents' beliefs and intentions should be straightforward. It is also hoped that future work will address more aspects of context than the two detailed and implemented in this paper, namely, the immediate and background context. Furthermore, the sample implementation of the extensions suggested in this paper serves as an example of how the extensions to DRT can be implemented. One way of developing this implementation is to incorporate it into a dialogue system which aims to achieve a more balanced approach to the semantic/pragmatic interface in representing linguistic phenomena.

References

Kamp, H. and Reyle, U. 1993. *From Discourse to Logic: Introduction to Model Theoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Boston, Dordrecht: Kluwer.

van Eijck, J. and Kamp, H. 1997. 'Representing Discourse in Context'. In: J. van Benthem and A. Ter Meulen (Eds.). *Handbook of Logic and Language*. pp. 179–237. Amsterdam: Elsevier.

Simons, M. 2003. 'Presupposition and Accommodation: Understanding the Stalnakerian Picture'. *Philosophical Studies* 112, pp. 251–278.

Werth, P. 1999. *Text Worlds: Representing Conceptual Space in Discourse*. New York: Longman.

RETROSPECTIVE ANALYSIS OF COMMUNICATION EVENTS - Understanding the Dynamics of Collaborative Multi-Party Discourse.

Andrew J. Cowell **Jereme Haack** **Adrienne Andrew**

Rich Interaction Environments
Pacific Northwest National Laboratory
Richland, WA 99336
{andrew.cowell | jereme.haack | adrienne.andrew}@pnl.gov

Abstract

This research is aimed at understanding the dynamics of collaborative multi-party discourse across multiple communication modalities. Before we can truly make significant strides in devising collaborative communication systems, there is a need to understand how typical users utilize computationally supported communications mechanisms such as email, instant messaging, video conferencing, chat rooms, etc., both singularly and in conjunction with traditional means of communication such as face-to-face meetings, telephone calls and postal mail. Attempting to understand an individual's communications profile with access to only a single modality is challenging at best and often futile. Here, we discuss the development of RACE – Retrospective Analysis of Communications Events – a test-bed prototype to investigate issues relating to multi-modal multi-party discourse. We also examine future avenues of research for further enhancing our prototype and investigating this area.

1 Introduction

Communication is the heart of what makes us social creatures. Today, we have a myriad of technologies that allow us to communicate in ways our forefathers could never have imagined. Computationally supported modalities such as email and instant messaging have had immeasurable effect on the way we work, play and generally interact with those in our lives. Being able to understand how individuals communicate, the methods they use, their personal preferences etc., and are all part of a field called anthroposemiotics[1]. This field looks to uncover the mystery behind how we communicate with ourselves (intrapersonal communication), with others (interpersonal communication), within groups (group dynamics) and across cultures (cross-cultural communication). While there is a great deal of literature in these fields, there are few operational applications that allow for true hands-on investigation.

Perhaps nowhere is the application of this field more important than in the field of intelligence analysis. Intelligence analysts must make sound judgments, coherently constructed from scattered heterogeneous fragments of information while being faced with significant time constraints. The information they use is rarely complete, often unreliable and usually temporally and spatially diverse. These dimensions need to be aligned and the information understood to enable the analyst to recognize sequences of inter-related events and hypothesize about future actions.

Our aim has been to aid the analyst by researching, designing and implementing a test-bed for the investigation of collaborative, multi-party discourse. The focus is on reducing the complexity of analyzing communications data through a triage process; from a large corpus to a small handful of relevant conversations to finally a highly detailed view of one or two conversations, with incorporated socio-behavioral dimensions. Below we present our design methodology and discuss the latest version of the prototype.

[1] http://en.wikipedia.org/wiki/Human_communication

Proceedings of the Analyzing Conversations in Text and Speech (ACTS) Workshop at HLT-NAACL 2006, pages 62–70,
New York City, New York, June 2006.

2 Method

The design methodology we used included a review of the literature, followed by in-depth focus group discussions with working analysts to determine requirements. Following the group session, a participatory design process was used to gather more information from our user group, leading to a set of sketches that were used in the initial prototype implementation. From these an initial prototype was created. The test-bed is currently in its second phase of implementation that includes the integration of components developed under other auspices into the RACE environment. These include indicators of affect and social roles. Finally, we have built and collected a number of data sources that we intend to use to evaluate the system. We describe these stages in the following sections.

2.1 Prior Art

This effort began with a thorough literature review across the fields of ubiquitous computing, visualization, multi-party discourse and communications theory. A number of research systems with similarities to our goal were reviewed in order to be able to understand the landscape and determine where specific opportunities may lie. Here we discuss some of the systems (mainly research prototypes) that are available for reviewing communications data.

As both Internet communications and complex graphics capabilities have become more pervasive in modern computing, there has been much interest in visualizing conversations. Due to the ease of data capture with computationally supported communications, such communication modalities as email, chat, and forum/newsgroup threads appear to be the most researched. Several systems have represented vast, multi-threaded newsgroup or forum posts such as USENET. 'Loom' can represent the activity patterns of individuals relative to one another, helping to characterize individuals' participation and roles (Donath et al, 1999). In another view, linked posts are graphed to represent threads, characterizing the newsgroup as a whole. 'Discourse Diagrams' describes newsgroups with semantic graphs of related concepts, and also graphs people's connectedness to one another in social networks (Sack, 2000). 'Conversation Thumbnails' uses an overview/detail display to contextualize a user's post in the group as a whole while it is being composed (Wattenberg and Millen, 2003). 'PeopleGarden' represents each individual participant as a composite of their history of posting. Having all participants represented in the same screen provides insight into the dynamics of the group as a whole across its recorded history, although there is no way to track connections between individuals or threading (Xiong and Donath, 1999).

In RACE, the topics of a multitude of conversations are explored by an analyst looking for both episodic and social information. Through an iterative filtering process, the analyst examines individual conversations. Like the newsgroup visualizations above, the goals are (in addition to a general desire to understand what is going on) to determine an individual's social role and dynamic of the group, but the concept of "conversations" is more granular. Whereas the newsgroup visualizations may represent hundreds or even thousands of users and conversation threads, the detailed visualization in RACE's final screen represents a single discourse with as few as two people. Thus, the systems above deal with a higher level of abstraction and do not convey information on "lurkers" who may read but not post, emotional qualities of contributions, or the temporal information present in synchronous communication. RACE has the additional goals of denoting presence, affect, and what Viegas and Donath call "negotiation of conversational synchrony" (1999).

Research on chat room conversation has produced some interesting visualizations that start to deal with these concepts. The 'Babble' system both facilitates and visualizes synchronous and asynchronous chat (Erickson and Laff, 2001). Users are represented as colored dots on a social proxy called a 'cookie'. The more interactions they have with the system, whether posting or only reading, the more central they become in the visualization. With inactivity, the dots move slowly back out to the periphery of the cookie, conveying information about presence and activity level. 'Chat Circles' is designed for synchronous chat and creates a strong sense of location by situating participants (represented as colored circles) in a large 2D space and only allowing them to see the text posted by others positioned nearby (Viegas and Donath, 1999). The circles expand to encompass posted text and shrink when ample time to

read the utterance has passed. Even people who are idling or only listening are represented spatially so others can see them. People can position their circles to avoid the 'noise' of unrelated conversations (as one could do at a cocktail party) or signify whom they are addressing. Each post leaves a cumulative translucent trace, indicating how long the poster has been there and how active they have been. Thus, group dynamics such as a group conversation fragmenting into smaller ones, relative verbosity, and relative position are available for interpretation.

While each of the systems above is designed for a particular modality, RACE integrates email, instant messaging, text messaging, phone conversations and teleconferences, in-person meetings in addition to chat or newsgroup participation. The goal is to get a more holistic sense of an individual throughout their discrete conversations and communication methods. As a post-hoc analysis tool, RACE aids the analyst by adding system interpretations of affect and social dynamics to the information represented in the prior art. It should be noted that this effort violates one of Erickson's six claims about social visualization: "Portray actions, not interpretation… users understand the context better than the system ever will" (2003). We agree in theory, but the needs of our analysts differ from those of a contributor to the conversation. Content-driven interpretations of group dynamics, affect, and social role complement full-text transcripts of the conversations, providing shortcuts to insight. Below we discuss further the requirements of our user group.

2.2 Requirements Elicitation

To ensure our research was applicable to our organization's missions and fulfilled the requirements and expectations of our user group, we enlisted the help of four analysts to determine specific requirements. These were to be our subject matter experts (SME's). Through interactions with our SME's we determined that while it is important to being able to understand a single conversation in time, it is just as, if not more, important to be able to comprehend the stream of conversations that occurs over longer periods, related to the same topic. For example, it is important to be able to intercept, process, and analyze a discussion between two individuals talking about making a homemade bomb, but it is even more important to place such a discussion within the context of the set of communications leading to an understanding of the overarching plot. Such review can provide additional information that could be invaluable to the analyst. Other requirements identified as part of these sessions included:

- The system should allow the analyst to get back to original source documents and be able to review the provenance.
- The system should allow the analyst to annotate the communication events.
- Consider the use of color for note taking and marking modalities.
- The system should allow the analyst to highlight conversation fragments (i.e., small parts of a larger conversation that are considered important).
- The system should provide basic translation mechanisms for foreign language support as well as provide some form of lexicon for terms that fall outside an analyst's field of expertise.
- The system should be able to import and export conversation fragments using common formats. The system should allow multiple analysts to work collaboratively within the same workspace.
- The system should allow the analyst to customize the environment to their preferences.

In addition to an informal list of requirements, a great deal of brainstorming was performed during this session. Following a participatory design process, system designers worked with SME's to put together a work process and some initial sketches of the overall system that could be fed into the implementation stage.

The process was designed so that the analyst could (Figure 1) interact with the conversation corpus available to them (potentially produced as a result of a search), viewing the conversations as dots, clustered around major topics. This view could be filtered based on time period, participants involved and communications modality used.

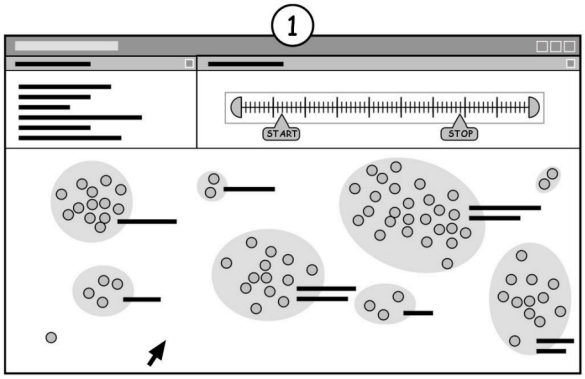


Figure 1: Sketch of the Corpus View.

On selecting a subset of conversations to review further (Figure 2) the analyst moves through to a second screen (the sequence view) where they can analyze the conversations in relation to when they occurred (the view is reminiscent of Microsoft Project's Gant view).

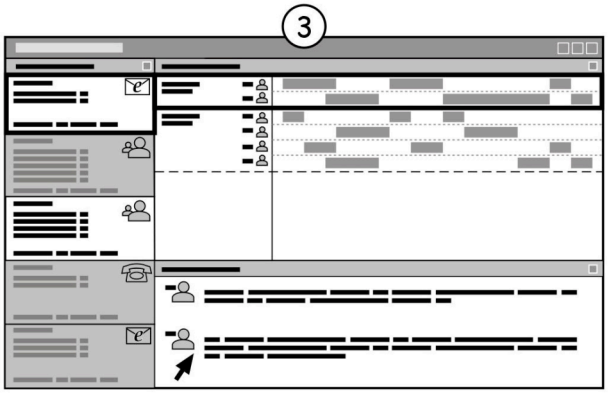


Figure 2: Sketch of the Sequence View.

While icons and text will continue to depict the modality the conversation utilized, the focus at this level is of fusing the conversations to build a sequenced stream of communications traffic so the underlying thread or purpose can be understood. Finally (Figure 3), conversations of specific interest to the analyst can be pursued in further detail in a third screen, called the 'detail view'. Here, the full transcript is displayed and can be 'played' utterance by utterance in real time. As each utterance is reached, a text-to-speech engine speaks the words, while a number of visual representations indicate social constructs such as social roles and the dynamics between the individuals.

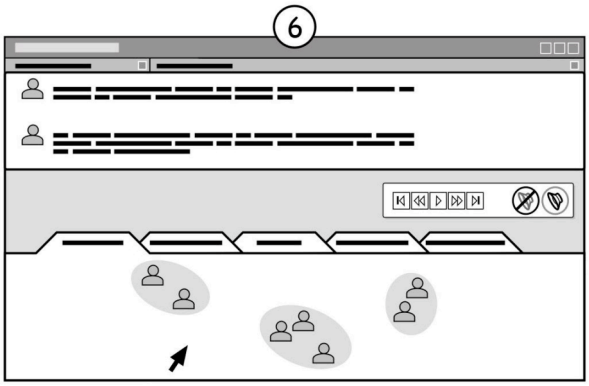


Figure 3: Sketch of the Detail View.

2.3 Implemented Prototype

Using a participatory design process, informed by the sketches and requirements of our analysts and the limitations of current research systems, we implemented a three-screen prototype analytical environment that allows a user to visualize a large corpus of communications events (Figure 4).


Figure 4: Analyst using the RACE Environment.

The environment can run on three screens simultaneously, be split across three panes (useful for performing analysis on large displays like wall-mounted plasma displays) or on a single screen with the use of a window manager seen in the top right of each view.

For the 'corpus view' (left hand screen, Figure 5) we customized some commercially available visualization software to present the conversation corpus, clustered by topic. Zooming in to individual items brings up metadata about that specific conversation. The different modalities may also be represented by different icons or colors,

depending on the type of style sheet loaded. Filters currently available include the modality used, the participants involved and the time/date the conversation occurred (and shortcuts to selecting all or none, or the current inverse are also available). Finally, a navigation window ensures the user does not get lost when interacting with a massive data that is topically diverse.

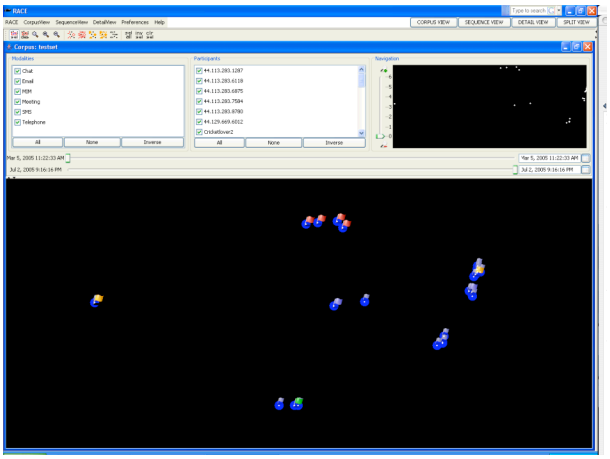


Figure 5: The Corpus View.

The 'sequence view' (Figure 6) is where we envision the majority of an analyst's time will be spent. It is here that they will review, in detail, a small subset of conversations that they found of interest in the corpus space. For example, in their exploration of the visualization, the analyst may find a group of discussions about a particular chemical substance. Knowing that this is relevant to a study they are performing, they simply drag a box around that subset and immediately those conversations are shown in the sequence view. Each conversation has an independent time line and can be zoomed out to show the entire conversation or zoomed in to see the individual utterances (these may also be accessed using tool-tips). The conversation titles on the left hand side of the screen can be unexpanded to show all the participants involved. Clicking on the participant opens a dialog box containing known information about that individual (including any known aliases and other names they may use online). A global timeline at the bottom of the screen shows where each conversation falls in sequence.

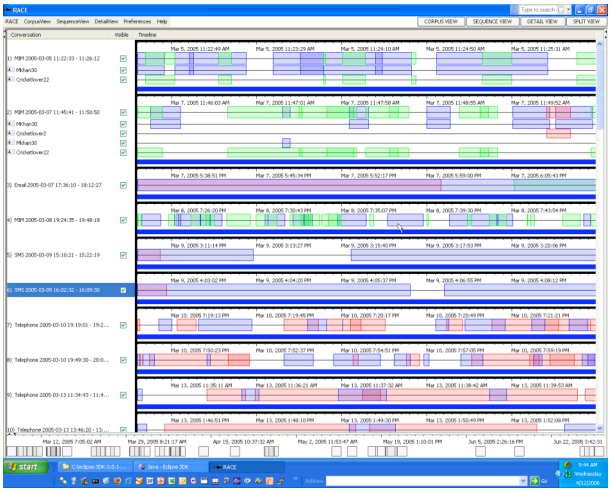


Figure 6: The Sequence View.

Once an important conversation is uncovered through the triage process, it can be selected for deeper investigation in the details view (Figure 7). This view can enable the analyst to see beyond the individual utterances. Utilizing other research performed at the Pacific Northwest National Laboratory, the details view enables the analyst to gain insight into an individual's opinion on the topics discussed. The transcript is color-coded to show the seven dimensions of affect (expression, power, ethics, attainment, skill, accomplishment and transactions), while a graph representation allows the analyst to compare individuals' affect against each other. In order to ingest the text in different ways, a 'text-to-speech' engine can be used to have the computer 'speak' the transcript. As it steps through the utterances, a group dynamics graphic (based on Erickson's Social Proxy) shows how the individuals relate to each other, highlighting those involved in the conversation and those that are idle. This view also provides a hierarchical view of the topics discussed with the ability to trigger a multi-dimensional visualization that maps participants to topics.

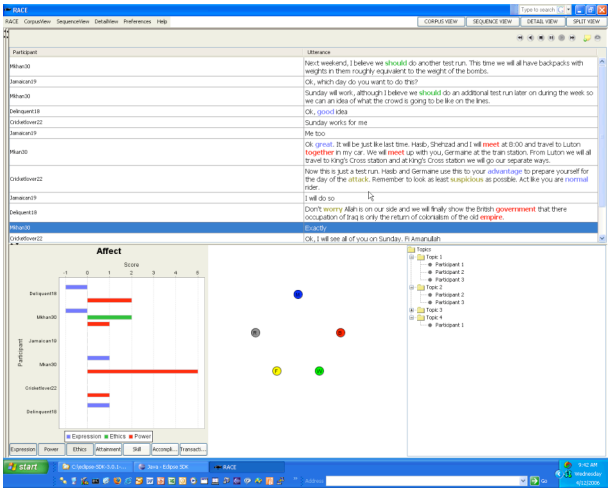


Figure 7: The Detail View.

3 Evaluation and Data Sets

In addition to the prototype system, an evaluation plan was developed. The current dataset being used to demonstrate the system was synthesized from news reports about the London bombings of 7th July 2005. The evaluation will use a new dataset build up from telephone transcripts from the regional August 14, 2003 blackout[2] to ensure any analysts used that were involved in the development of the prototype will not benefit from any potential learning effects. This data is made up of several participants involved in many different conversations. These characteristics are exactly what RACE was designed for. Another dataset is a transcript of a murder mystery held on a chat room. While there was only space for characters to interact, there are many different threads of conversation going on at once. This data set will be useful for exploring the social dynamic part of RACE. We hope to show who the conversational "drivers" were and explore what characteristics give someone away as hiding details they do not want other characters to discover.

4 Summary & Further Work

The ultimate goal of the RACE project is to assist analysts as they try to extract meaning from a myriad of sources. To this end, we started by talking with analysts themselves. This is in recognition of the fact that no matter how powerful a tool might seem to its developers, it is useless unless the end users actually adopt it. By working with analysts every step of the way, we are keeping that goal in sight.

RACE's design as a test bed enables other research to get in front of the analyst sooner. The quick insertion of the text affect work illustrates the capability to make functionality available to the user for evaluation. Showing an analyst a concrete example of an idea allows them to get a better understanding of it and an easier way to elicit feedback for future work.

While this is an exciting first step, there are many avenues of crucial research still to be performed. In many fields, having access to all the communications events that occurred is rare. Research needs to be performed to determine how best to enable the analyst to fill in these blanks. Potential approaches include hypothesized inference or the use placeholders.

Currently, the prototype analytical environment only processes and displays textual transcripts of communication events. This decision was made to handle textual content first so to ensure proof of principle prior to expending effort on the more challenging aspect of fusing video, audio, still images and text (VAST). Some effort has been expended on looking for suitable design metaphors that could aid an analyst in making sense of such diverse media (e.g., video production user interfaces such as Final Cut Pro) but more research, design and evaluation is required.

More effort needs to be expended on understanding how best to fuse different modalities of communication. Currently, a time-shifting approach is used to normalize an asynchronous email thread with similar-topic synchronous communications (e.g., telephone call, instant messaging session). This approach works but needs to be refined in order to be successful. At one level, the modality used is irrelevant – it is the essence of the event that is of primary concern. Being able to boil down the associated threads into one specific stream (e.g., multiple conversations across a number of modalities, all around the topic of plotting to explode a device at a particular location) is crucial in being able to support the analytical tradecraft and allow analysts to provide actionable intelligence to their superiors.

Conversations rarely keep to one single focused topic, and this can cause problems in the cluster visualization type approach used so far.

[2] http://www.nerc.com/~filez/blackout.html

Topic segmentation is a difficult research area and not one that we intend to pursue. There are at least three projects currently on the way at our institution that deal with this area and this work intends to utilize the fruits of those labors.

Finally, there are many elements of multi-party discourse that exist outside linguistic boundaries. The words we use, how often we make an utterance, etc., all speak to who we are as individuals. While some of this is obvious and can be observed with just a cursory review of a transcript of the source material, other elements are discrete and hidden. For example, conversational statistics can be recorded and used to determine an individual's level of engagement in a topic. Detection of familiarity (e.g., either by specific words not currently found in the present conversation or through the use of casual rather than formal speech) can indicate personal relationships between individuals in a dyad. Personality types can be inferred by markers indicative of leadership (e.g., number of interruptions performed/received, ability to change topic, use of power terms) or weaker, subversive roles (e.g., use of weak terms, submission of floor, deference to others). Analysts are rarely able to access such rich personality profiles of their subjects without performing an exhaustive analysis or calling in specialized help. While we are just beginning to integrate certain elements of social discourse, there are many other dimensions to be considered.

References

Donath, J., Karahalios, K., and Viegas, F. 1999. "Visualizing conversations," Proceedings of HICSS-32, Persistent Conversations Track. Maui, Hawaii, January 5-8.

Sack, W. 2000. "Discourse Diagrams: Interface Design for Very Large Scale Conversations," Proceedings of the HICSS-33, Persistent Conversations Track . Maui, Hawaii, January.

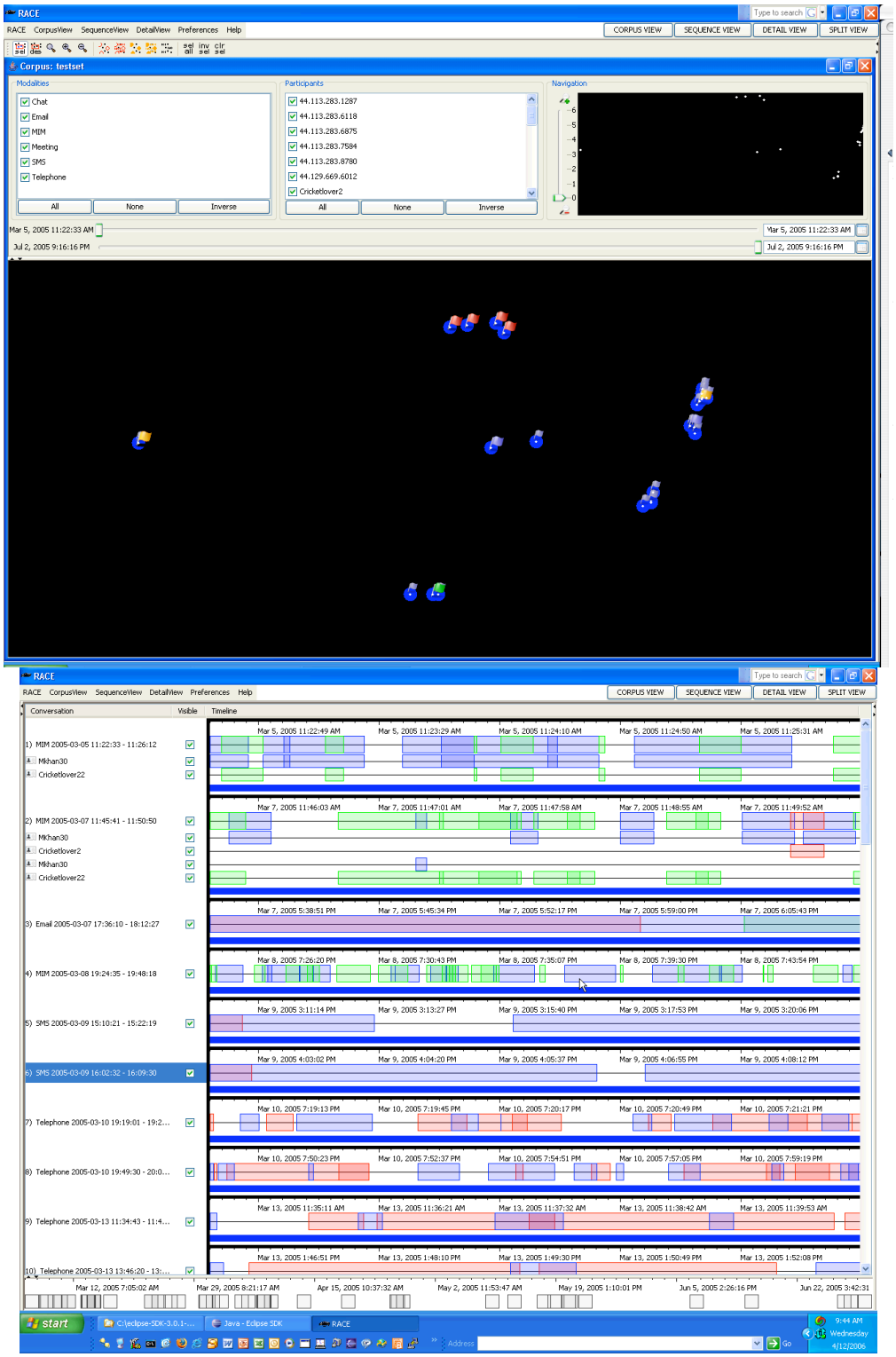
Wattenberg, M. M & Millen, D. R., (2003) Conversation Thumbnails for Large-Scale Discussions. . In Extended Abstracts – CHI 2003. Ft. Lauderdale, April 8-10.

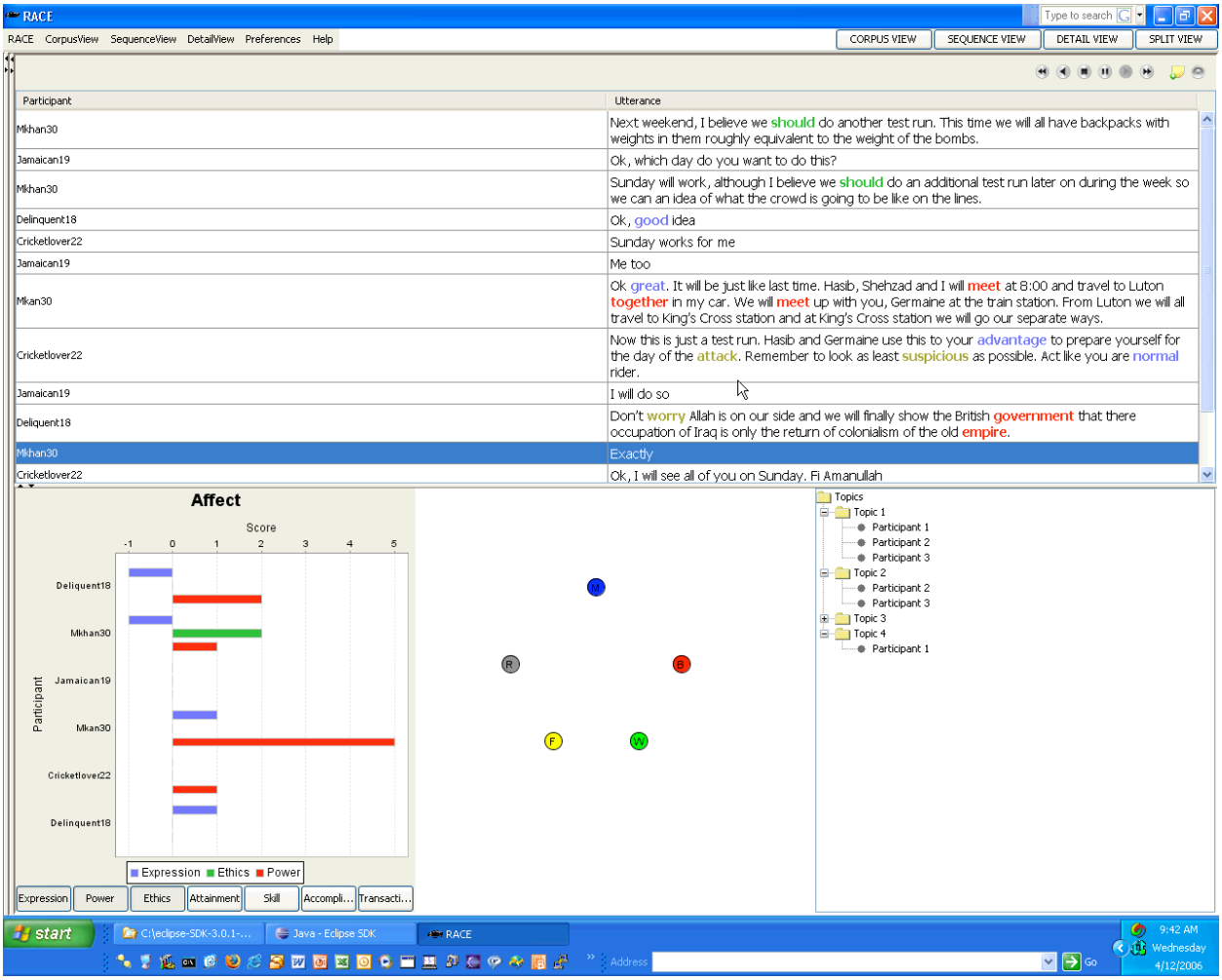
Xiong, R. and Donath, J. PeopleGarden: Creating Data Portraits for Users, in Proceedings of the 12 th Annual ACM Symposium on User Interface Software and Technology, New York: ACM, pp 37-44.

Viegas , F. B. & Donath, J. S. (1999). Chat Circles. CHI 1999 Conference Proceedings: Conference on Human Factors in Computing Systems. New York: ACM Press, 9-16.

Erickson T and Laff MR (2001) The design of the 'Babble' timeline: A social proxy for visualizing group activity over time. Extended Abstracts: Human Factors in Computing System (CHI 2001), 329-220.

Erickson, T. Designing Visualizations of Social Activity: Six Claims, The Proceedings of CHI 2003: Extended Abstracts. New York: ACM Press, 2003.


RACE
RACE CorpusView SequenceView DetailView Preferences Help
CORPUS VIEW
SEQUENCE VIEW
DETAIL VIEW
SPLIT VIEW
Corpus: testset
Modalities
Chat
Email
MIM
Meeting
SMS
Telephone
All
None
Inverse
Participants
44.113.283.1287
44.113.283.6118
44.113.283.6875
44.113.283.7584
44.113.283.8780
44.129.669.6012
Cricketlover2
Navigation
Mar 5, 2005 11:22:33 AM
Jul 2, 2005 9:16:16 PM
Conversation
Visible
Timeline
1) MIM 2005-03-05 11:22:33 - 11:26:12
Mkhan30
Cricketlover22
2) MIM 2005-03-07 11:45:41 - 11:50:50
Mkhan30
Cricketlover2
Mkhan30
Cricketlover22
3) Email 2005-03-07 17:36:10 - 18:12:27
4) MIM 2005-03-08 19:24:35 - 19:48:18
5) SMS 2005-03-09 15:10:21 - 15:22:19
6) SMS 2005-03-09 16:02:32 - 16:09:30
7) Telephone 2005-03-10 19:19:01 - 19:2...
8) Telephone 2005-03-10 19:49:30 - 20:0...
9) Telephone 2005-03-13 11:34:43 - 11:4...
10) Telephone 2005-03-13 13:46:20 - 13:...
Mar 12, 2005 7:05:02 AM
Mar 29, 2005 8:21:17 AM
Apr 15, 2005 10:37:32 AM
May 2, 2005 11:53:47 AM
May 19, 2005 1:10:01 PM
Jun 5, 2005 2:26:16 PM
Jun 22, 2005 3:42:31
start
Java - Eclipse SDK
RACE
9:44 AM
Wednesday
4/12/2006


RACE
RACE CorpusView SequenceView DetailView Preferences Help
CORPUS VIEW
SEQUENCE VIEW
DETAIL VIEW
SPLIT VIEW
Participant
Utterance
Mkhan30
Next weekend, I believe we should do another test run. This time we will all have backpacks with weights in them roughly equivalent to the weight of the bombs.
Jamaican19
Ok, which day do you want to do this?
Mkhan30
Sunday will work, although I believe we should do an additional test run later on during the week so we can an idea of what the crowd is going to be like on the lines.
Delinquent18
Ok, good idea
Cricketlover22
Sunday works for me
Jamaican19
Me too
Mkan30
Ok great. It will be just like last time. Hasib, Shehzad and I will meet at 8:00 and travel to Luton together in my car. We will meet up with you, Germaine at the train station. From Luton we will all travel to King's Cross station and at King's Cross station we will go our separate ways.
Cricketlover22
Now this is just a test run. Hasib and Germaine use this to your advantage to prepare yourself for the day of the attack. Remember to look as least suspicious as possible. Act like you are normal rider.
Jamaican19
I will do so
Deliquent18
Don't worry Allah is on our side and we will finally show the British government that there occupation of Iraq is only the return of colonialism of the old empire.
Mkhan30
Exactly
Cricketlover22
Ok, I will see all of you on Sunday. Fi Amanullah
Affect
Score
-1 0 1 2 3 4 5
Deliquent18
Mkhan30
Jamaican19
Mkan30
Cricketlover22
Delinquent18
Participant
Expression Ethics Power
Expression
Power
Ethics
Attainment
Skill
Accompli...
Transacti...
M
R
B
F
W
Topics
Topic 1
Participant 1
Participant 2
Participant 3
Topic 2
Participant 2
Participant 3
Topic 3
Topic 4
Participant 1
start
C:\eclipse-SDK-3.0.1-...
Java - Eclipse SDK
RACE
Address
Go
9:42 AM
Wednesday
4/12/2006

Author Index