# Developing a Method of Dialogue Segmentation

Emma Conner, Yuan Du, and Rebecca Passonneau
Department of Computer Science
Columbia University
Distributed REU

August 4, 2012

**Abstract**

Dialogue segmentation is important for applications using flawed speech recognition results. In this paper, we present a variety of potentially-useful features for automatic extraction of discourse segments. We then test these segments in a number of experiments, using various sampling techniques, various representations of segments, and various machine learning methods. We determine that a method of oversampling to correct for data imbalance and a representation of segments as the presence of a boundary between two segments perform the best.

## 1 Introduction

**Motivation** This project intends to develop automatic methods to divide dialogues into discourse segments. Discourse segments are defined as multi-utterance units in which the speakers involved are engaged in a single communicative task. Previous research has shown the validity of similar intention-based segments in monologues, and the potential to develop methods of automatically extracting them [5], but little work has been done to develop similar methods for multi-speaker dialogues.

It is hypothesized that dialogues will be able to be segmented into high-coherence and low-coherence segments, and that in high-coherence segments,

in which speakers have more interest and confidence in what they are saying, they will speak with a larger pitch and intensity range, and with fewer disfluencies, all factors which are shown to improve speech recognition [2]. Thus, besides simply providing insight into dialogue structures, dialogue segmentation should assist with applications that use data from imperfect speech recognizers, as we will be able to put more confidence in some segments and less in others.

**The Babel Project** This research was done as a part of the Babel project, a project to develop new methods of rapidly developing speech recognition software for low-resource languages. Current speech recognition software takes months to develop, and hours of good quality, hand-tagged speech data in that language. Therefore, good speech recognition software is currently available in only a handful of languages. To correct this, the Babel project provides participants with very little speech data of imperfect quality. Within the limitations of the Babel project, it is important to develop ways to help applications use potentially imperfect results.

In Section 2 we will present the dataset used for this project and the features extracted. Section 3 describes the machine learning experiments performed, and presents the results, and Section 4 presents conclusions and describes future pathways for this project.

# 2 Methods

**Dataset** This project uses data from the Loqui corpus, which consists of 82 dialogues between librarians and patrons at the Heiskell Braille and Talking Book Library of New York City. This library provides books in braille and audiobooks to patrons, and most transactions occur over the phone or through the mail. The calls were previously transcribed and annotated with Dialogue Functional Units and with Task Success and Cost Annotation, of particular relevance to this project. The tasks annotated correspond to segments of related utterances which focus around a given dialogue task, such as a patron requesting to check out a book, or a librarian obtaining the patron's address.

**Features** Many features were proposed, and evaluated for ease of implementation and potential usefulness. For more difficult to collect features, we tested usefulness by randomly selecting four dialogues, extracting the feature in question by hand, and calculating precision and recall. If these values were too low, the features were deemed not worth the effort to be included in the initial set of features. For example, we originally hypothesized that question word order, or a final rise in pitch, might be useful due to the frequency with which segments in a few dialogues began with questions, but upon examination, the pattern was not born out among all sample dialogues, and the feature was not collected.

While some proposed features remain for future implementation, a total of 33 features were eventually selected and collected, from the broad categories of acoustic features, pause features, length reduction features, and lexical features. All features are described in table 1 below.

| Feature Name | Description |
|---|---|
| Acoustic Features | |
| $Pitch_{MIN}$ | Minimum pitch of utterance $i$ |
| $Pitch_{MAX}$ | Maximum pitch of utterance $i$ |
| $Pitch_{MEAN}$ | Mean pitch of utterance $i$ |
| $Pitch_{STDEV}$ | Standard deviation of pitch in utterance $i$ |
| $Pitch_{RANGE}$ | Range in pitch over utterance $i$: $Pitch_{MAX}(i) - Pitch_{MIN}(i)$ |
| $Pitch_{CHANGE}$ | Change in mean pitch from previous utterance: $Pitch_{MEAN}(i) - Pitch_{MEAN}(i-1)$ |
| $Intensity_{MIN}$ | Minimum intensity of utterance $i$ |
| $Intensity_{MAX}$ | Maximum intensity of utterance $i$ |
| $Intensity_{MEAN}$ | Mean intensity of utterance $i$ |
| $Intensity_{STDEV}$ | Standard deviation of intensity in utterance $i$ |
| $Intensity_{RANGE}$ | Range in intensity over utterance $i$: $Intensity_{MAX}(i) - Intensity_{MIN}(i)$ |
| $Intensity_{CHANGE}$ | Change in mean intensity from previous utterance: $Intensity_{MEAN}(i) - Intensity_{MEAN}(i-1)$ |
| Pause Features | |
| $Pause_{DURT}$ | Total duration of pauses (in seconds) in utterance $i$ |
| $Pause_{RATIO}$ | Ratio of seconds of pauses to total length of $i$: $Pause_{DURT}/LR_1$ |
| $Pause_{BEG}$ | Presence of a transcribed pause tag at the beginning of utterance $i$ |
| $Pause_{END}$ | Presence of a transcribed pause tag at the end of utterance $i$ |
| $Pause_{MID}$ | Presence of a transcribed pause tag at the middle of utterance $i$ |
| $FP_{BEG}$ | Presence of a filled pause at the beginning of utterance $i$ |

| | |
|---|---|
| $FP_{END}$ | Presence of a filled pause at the end of utterance $i$ |
| $FP_{MID}$ | Presence of a filled pause in the middle of utterance $i$ |
| Length Reduction Features | |
| $LR_1$ | Absolute duration in seconds of utterance $i$ |
| $LR_{1Normalized}$ | $LR_1(i)/average(LR_1(j)), j \in$ utterances by speaker of $i$ |
| $LR_2$ | Number of words in utterance $i$ |
| $LR_{2Normalized}$ | $LR_2$ / Average number of words over all utterances by the speaker of $i$ |
| $LR_3$ | Average seconds per word in $i$: $LR_1/LR_2$ |
| $LR_{3Normalized}$ | $LR_{1Normalized}/LR_{2Normalized}$ |
| $LR_4$ | Average word length, in characters, over all words in $i$ |
| $LR_5$ | Length of longest word in $i/LR_4$ |
| Lexical Features | |
| $LR6_1$ | Uncommon character feature 1: Average frequency of use in English of all characters in $i$ |
| $LR6_2$ | Uncommon character feature 2: Total number of characters in i which have frequency of use in English of less than a certain threshhold |
| $IR$ | Information reduction: Number of words remaining in $i$ after removing discourse maintenance words, defined as words that appear more frequently in spoken English than in written English. Frequencies taken from the American National Corpus [4]. |
| $PN_1$ | Proper noun feature 1: number of proper nouns in utterance $i$ / total number of words in the utterance |
| $PN_2$ | Proper noun feature 2: presence of a proper noun in utterance $i$ never before seen in the dialogue |

Table 1. Names and descriptions of all 33 features extracted.

In addition, three types of labels were collected, to determine the most natural representation of a segment. First, we collected segment-initial labels, in which the first utterance of each segments were marked, in order to capture features that might occur when a new segment occurs; second, we collected segment-final labels, in which the last utterance of each segment is marked, to capture features which might occur at the end of segments; third, we created a representation of the features as a window of two utterances, and labeled whether or not there was a segment boundary between them, to capture

changes between the end and beginning of a segment.

# 3 Experiments

**Setup** A total of 25 machine learning experiments were performed in the WEKA machine learning toolkit [3]. Experiments were performed using C4.5 Decision Trees, Naive Bayes, and logistic regression for each set of labels.
Initial tests were performed using a subset of 11 of the features and only the decision tree and Naive Bayes method. These revealed that the severe imbalance of the dataonly about 10% of instances are in the positive classmade learning difficult. Nearly all the instances could be labeled as negative with relatively good overall precision and recall, leaving extremely poor recall in the positive class. To correct this, in the final set of experiments we used all three learning methods and all three labeling methods with a dataset in which the positive class was oversampled using the SMOTE oversampling method, as described in [1], and experiments with all three learning methods and segment-initial labels with data in which the negative class was undersampled by randomly selecting and deleting instances of the negative class. Experiments with undersampled data and the other two label types will be performed in future work.

Note that the oversampled and undersampled experiments were tested using cross-validation with the over- and under-sampled datasets, respectively. Tests of the resulting models on datasets with the original data balance remain to be done.

**Results** The results of the previously described experiments are shown below in table 2. For each experiment, we calculate average precision, recall, and F-measure. For the purpose of presenting the problem described above, of poor recognition of the positive class due to data imbalance, these measurements are presented for the positive class alone as well.

| Learner | Precision | Recall | F-Measure | Positive class Precision | Positive class Recall | Positive class F-measure |
|---|---|---|---|---|---|---|
| Original Data Balance | | | | | | |
| Segment-initial labels | | | | | | |
| J48 | 0.842 | 0.877 | 0.86 | 0.323 | 0.139 | 0.194 |
| NB | 0.861 | 0.426 | 0.508 | 0.137 | 0.825 | 0.234 |
| LR | 0.831 | 0.893 | 0.844 | 0.304 | 0.007 | 0.014 |
| Segment-final labels | | | | | | |
| J48 | 0.852 | 0.890 | 0.859 | 0.427 | 0.104 | 0.167 |
| NB | 0.819 | 0.861 | 0.837 | 0.165 | 0.075 | 0.104 |
| LR | 0.822 | 0.893 | 0.844 | 0.222 | 0.002 | 0.004 |
| Inter-utterance labels | | | | | | |
| J48 | 0.846 | 0.867 | 0.855 | 0.315 | 0.214 | 0.255 |
| NB | 0.857 | 0.687 | 0.745 | 0.192 | 0.604 | 0.291 |
| LR | 0.858 | 0.894 | 0.852 | 0.524 | 0.044 | 0.082 |
| Oversampled Data | | | | | | |
| Segment-initial labels | | | | | | |
| J48 | 0.891 | 0.891 | 0.891 | 0.908 | 0.878 | 0.89 |
| NB | 0.672 | 0.635 | 0.607 | 0.6 | 0.886 | 0.715 |
| LR | 0.726 | 0.726 | 0.726 | 0.727 | 0.754 | 0.74 |
| Segment-final labels | | | | | | |
| J48 | 0.864 | 0.863 | 0.863 | 0.889 | 0.839 | 0.863 |
| NB | 0.687 | 0.673 | 0.664 | 0.643 | 0.83 | 0.724 |
| LR | 0.706 | 0.706 | 0.706 | 0.708 | 0.736 | 0.721 |
| Inter-utterance labels | | | | | | |
| J48 | 0.877 | 0.877 | 0.877 | 0.883 | 0.88 | 0.881 |
| NB | 0.775 | 0.772 | 0.772 | 0.803 | 0.741 | 0.771 |
| LR | 0.772 | 0.772 | 0.772 | 0.772 | 0.794 | 0.783 |
| Undersampled data | | | | | | |
| Segment-initial labels | | | | | | |
| J48 | 0.597 | 0.596 | 0.597 | 0.582 | 0.587 | 0.584 |
| NB | 0.621 | 0.561 | 0.513 | 0.527 | 0.889 | 0.662 |
| LR | 0.658 | 0.658 | 0.658 | 0.654 | 0.622 | 0.637 |

Table 2. Results of all 25 experiments.

The oversampled decision tree with inter-utterance labeling performs best overall, and in general, inter-utterance labeling performs best. Little difference is apparent between segment-initial and segment-final labeling. This suggests that a segment boundary is not best represented as absolute values of features, but as a change between utterances. Oversampling performs better than both undersampling and the original data balance; undersampling provides little improvement over the original data balance, perhaps because so much data is lost that potentially useful features are obscured.

# 4 Future work

While initial results are promising, and it seems that there are features that can predict a change in communicative tasks by speakers, work should still be done to refine results further. More features should still be explored, and other learning methods examined to see what performs the best.

Additionally, this work must still be adapted to work for the Babel datasets. First, the structure of the dialogues are different; in the Loqui dialogues, participants are having natural conversations addressing a specific taskchecking out books, most frequentlywhile in the Babel datasets the calls are artificial and have no particular task they are addressing. The domain-independency of these features therefore must still be examined. In addition, the dialogues in the Loqui corpus were human-transcribed; it must still be tested on noisy data from speech recognizers. Finally, the language independency of these features have not been examined; it is possible that they will perform worse in other language.

# References

[1] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. Smote: Synthetic minority over-sampling technique. *J. Artif. Intell. Res. (JAIR)*, 16:321–357, 2002.

[2] Sharon Goldwater, Daniel Jurafsky, and Christopher D. Manning. Which words are hard to recognize? prosodic, lexical, and disfluency factors that

increase speech recognition error rates. *Speech Communication*, pages 181–200, 2010.

[3] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11(1):10–18, November 2009.

[4] N. Ide and K Suderman. The american national corpus first release. In *Proceedings of the Fourth Language Resources and Evaluation Conference*, pages 1681–84, Lisbon, 2004. LREC.

[5] Rebecca J. Passonneau and Diane J. Litman. Discourse segmentation by human and automated means. *Comput. Linguist.*, 23(1):103–139, March 1997.