ON THE SELECTION OF PROSODIC FEATURES FOR LANGUAGE MODELING

ALEJANDRO VEGA

Department of Computer Science

APPROVED:

Nigel Ward, Chair, Ph.D.

David Novick, Ph.D.

Jon Amastae, Ph.D.

Benjamin C. Flores, Ph.D.
Dean of the Graduate School

ON THE SELECTION OF PROSODIC FEATURES FOR LANGUAGE MODELING

by

ALEJANDRO VEGA, B.S.

THESIS

Presented to the Faculty of the Graduate School of

The University of Texas at El Paso

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE

Department of Computer Science

THE UNIVERSITY OF TEXAS AT EL PASO

December 2012

Acknowledgments

I would like to give thanks to all my family and friends. Without their support all these years, I would have never gotten to where I am right now. I want to thank my family for their sacrifices and patience. I want to thank my friends for their support and company during the good and tough times.

I want to give my deepest thanks to Nigel Ward for all his help as an advisor and mentor. I thank David Novick for being a mentor and teacher and for all the helpful feedback all these years. I thank Jon Amastae for all the helpful comments and being my committee member.

I would like to give deep thanks to Shreyas Karkhedkar for his help with the Respond features, for his help being someone I could always talk to when I had trouble, and his help as one of my best friends during this process.

This work was supported in part by NSF Award IIS-0914868.

Abstract

Previous studies show that immediate and long range prosodic context provide beneficial information when applied to a language model. However, the fact that some features provide more information to the prediction task should be considered. If the information contribution of each feature can be determined, then a well-crafted feature set can be built to improve the performance of a language model.

In this study, I measure the contribution of different prosodic features to a baseline trigram model. Using this information, it should be possible to build a language model that uses the most informative resources and ultimately performs better than a language model that includes prosodic information naively. Using this information, I build a prosodic feature set of 103 prosodic features from past and future context computed for both speaker and interlocutor. Principal component analysis is applied to this feature set to build a model that achieves a 25.9% perplexity reduction relative to a tri-gram model. However, this model falls short of performance improvements achieved by a similar model without proper feature selection by –1.2%.

Table of Contents

Page

Acknowledgments iii

Abstract iv

Table of Contents v

List of Tables viii

List of Figures ix

Chapters

1 Introduction 1

1.1 Motivation 2

1.1.1 Language Modeling 2

1.2 Thesis Statement 3

2 Related Work 4

2.1 Prosody and Language Modeling 4

2.2 Conversational Speech Applications 5

2.3 Summary 6

3 Methodology 7

3.1 Prosodic Features 7

3.1.1 Feature Calculation 7

3.1.2 Using Non-adjacent Features 10
3.1.3 Using Longer Context Features 11
3.2 Use in a Language Model 12
3.2.1 Combination with N-grams 13
3.3 Evaluation Scheme 15
3.3.1 Corpus 15
3.3.2 Performance evaluation 15
4 Initial Observations 17
5 Feature Evaluation 24
5.1 Non-adjacent Feature Models 24
5.2 Longer-Width Window Feature Models 30
5.2.1 Past speaker features 30
5.2.2 Future speaker features 34
5.2.3 Past interlocutor features 39
5.2.4 Future interlocutor features 43
5.3 Feature Selection 47
5.3.1 Speaker Features 48
5.3.2 Interlocutor Features 52
5.3.3 Selected Features 56
6 Principal Component Language Model 57
6.1 PC Language Models 58

7 Discussion 62

7.1 Model Benefit Analysis 62

7.2 Dimension Analysis 64

8 Conclusions and Future Work 72

8.1 Improvements 73

8.2 Future Work 73

References 75

Curriculum Vitae 78

List of Tables

3.1 Prosodic Feature Definition 8

6.1 Top 15 components and their respective perplexity reductions to the trigram baseline. 59

6.2 Perplexity reduction for combined PC model. 60

6.3 Perplexity reduction for tuned combined PC models. 61

7.1 Examples of mock-quoting *oh*. 63

7.2 Example utterance of model failure. 64

7.3 Example for dimension 89: Low dimension values. 66

7.4 Example for dimension 89: High dimension values. 66

7.5 Example for dimension 89: Low dimension values. 67

7.6 Example for dimension 89: High dimension values. 68

7.7 Example for dimension 99: Low dimension values. 68

7.8 Example for dimension 99: High dimension values. 69

7.9 Example for dimension 10: Low dimension values. 70

7.10 Example for dimension 10: High dimension values. 70

7.11 Example for dimension 3: Low dimension values. 71

7.12 Example for dimension 3: High dimension values. 71

List of Figures

3.1 Offset prosodic features. 10

3.2 Feature calculation over increasing window sizes. 12

4.1 S-Ratios for word *I*. 18

4.2 S-Ratios for word *and*. 18

4.3 S-Ratios for left pitch height context. 20

4.4 S-Ratios for left pitch range context. 20

4.5 S-Ratios for left speaking rate context. 21

4.6 Offset S-Ratio Mean Absolute Difference 22

5.1 Perplexity reduction for non-adjacent volume models. 25

5.2 Perplexity reduction for non-adjacent pitch height models 26

5.3 Perplexity reduction for non-adjacent pitch range models 28

5.4 Perplexity reduction for non-adjacent speaking rate models 29

5.5 Perplexity reduction for past volume models 31

5.6 Perplexity reduction for past pitch height models 32

5.7 Perplexity reduction for past pitch range models 33

5.8 Perplexity reduction for past speaking rate models 34

5.9 Perplexity reduction for future volume models 35

5.10 Perplexity reduction for future pitch height models 36
5.11 Perplexity reduction for future pitch range models 37
5.12 Perplexity reduction for future speaking rate models 38
5.13 Perplexity reduction for interlocutor past volume models 40
5.14 Perplexity reduction for interlocutor past pitch height models 41
5.15 Perplexity reduction for interlocutor past pitch range models 42
5.16 Perplexity reduction for interlocutor past speaking rate models 43
5.17 Perplexity reduction for interlocutor future volume models 44
5.18 Perplexity reduction for interlocutor future pitch height models 45
5.19 Perplexity reduction for interlocutor future pitch range models 45
5.20 Perplexity reduction for interlocutor future speaking rate models 46
5.21 Speaker past volume features 49
5.22 Speaker future volume features 50
5.23 Speaker past pitch height features 50
5.24 Speaker future pitch height features 50
5.25 Speaker past pitch range features 51
5.26 Speaker future pitch range features 51
5.27 Speaker past speaking rate features 52
5.28 Speaker future speaking rate features 52
5.29 Interlocutor past volume features 53
5.30 Interlocutor future volume features 53

5.31 Interlocutor past pitch height features 54

5.32 Interlocutor future pitch height features 54

5.33 Interlocutor past pitch range features 55

5.34 Interlocutor future pitch range features 55

5.35 Interlocutor past speaking rate features 56

5.36 Interlocutor future speaking rate features 56

Chapter 1

Introduction

Today, automatic speech recognition (ASR) systems are widely used. Once limited to dictation systems, ASR is a fundamental part of many systems with speech as a mode of interaction. From uttering simple voice commands to more complex voice queries, users and developers have many options for integrating ASR into applications. However, these systems still have limitations. Aside from signal quality and other problems, current systems still fail in recognizing words, ultimately leading to user frustration and a negative perception of such systems. Current systems utilize lexical context as their main source of information for word prediction, however this is sometimes not enough. In this study, I use prosody as an additional source of information for a language model. Prosody, or basically the way a person speaks in terms of features like intensity, pitch, and rate of speech, is present everywhere in both monologue and dialogue. This makes prosody a prime candidate as a source of information for ASR systems.

1.1 Motivation

This thesis focuses on discovering the information contribution of different prosodic features when applied to language modeling. While previous research shows that prosodic information does provide useful information to a language model (LM), these models incorporate prosodic information in an ad hoc manner. Thus, discovering the relative importance of features to language modeling enables the selection of the best features. Once selected, I apply the resulting features to a LM, with the aim of producing a model with better performance than one with prosodic information, but without proper feature selection.

1.1.1 Language Modeling

There are two ways to improve an ASR system: (1) improve the Acoustic Model (AM) or (2) improve the Language Model (LM). Conceptually, in an ASR system, the AM takes care of reducing the speech signal to a series of phonemes that are then matched to phonemes in a given dictionary. However different words may have similar or identical phoneme strings. To deal with this ambiguity, a LM assigns probabilities to all the words seen in training so that the ASR system can pick the most probable word in the LM list that matches the phoneme string generated by the AM. The probabilities assigned by the LM are probabilities of words estimated in various ways, including, most traditionally, from past lexical context using an n-gram model. In an n-gram model, the probability of a word is conditioned on the past n-1 word tokens. Statistics for such models are generated by looking at counts for all n-length strings found in a given training corpora that is often

domain specific.

1.2 Thesis Statement

The main hypothesis of this research is that a LM that incorporates the objectively most informative features will produce lower perplexity results than one that incorporates similar information selected naively. This is novel in that it focuses on finding the set of features that are important for application to a LM. This set of features can then later be applied to prediction endeavors for ASR.

The balance of this thesis is comprised of: Chapter 2 presents an overview of previous research and current state of the art models incorporating prosody. Chapter 3 describes the methodology used for computing the features used as well as the evaluation procedures. Chapter 4 presents a preliminary study of the behavior for the probabilities conditioned on non-adjacent features. Chapter 5 presents the characteristics of the best performing prosodic features. Chapter 6 presents the results of the features when applied to a LM and chapter 7 discusses the results. Chapter 8 lays out the conclusions and future work.

Chapter 2

Related Work

In this chapter I review past and current research on the use of prosody in language modeling and its applications on conversational speech domains.

2.1 Prosody and Language Modeling

This study is not the first of its kind to use prosody as a prediction mechanism for language modeling. Shriberg and Stolcke [6] surveyed the use of prosodic features as additional information for a LM. One important aspect that they advocate is that the dependence relationship between prosodic features and target classes (e.g., dialog acts, sentence segmentation, words, etc.) should be direct, not mediated by linguistically-hypothesized prosodic elements, such as tones. The target classes can be predicted directly through the use of prosodic features. Results for this study yielded performance improvements for the prediction tasks (2% relative reduction in word error rate (WER)), showing that using prosody was a promising avenue. Huang and Renals [5] took this concept further and produced a model that used prosody to predict words in an n-gram framework. Unfortunately, they found that their prosodic n-gram model could not deal with the large number of target

words to be predicted under this framework. They point out that this may be due to the lack of characteristic prosodic patterns for each word. While these may be adequate for prediction of small target class sets (e.g., dialog acts, boundary segmentation, etc.), the patterns may not be adequate to differentiate between large sets of words. However, this approach still yielded modest perplexity and WER improvements over a baseline model. These and other studies [1, 2, 8] have shown that the use of prosody as additional information can be beneficial to LMs. However, one thing these studies share in common is that they are focused on genres (notably radio broadcasts) and languages (notably Hungarian) where prosody has a set rhythmic pattern tied to the words spoken. The lack of characteristic prosodic patterns described in [5] may be attributable to the atypicality of such domains. Thus, if the integration of prosody to language modeling can be extended to a different dialog domain with higher variability in prosodic patterns, then prosody may be able to provide more beneficial information for the prediction task.

2.2 Conversational Speech Applications

One such dialog domain is conversational speech. In this domain, there is a high variability in the prosodic patterns observed due to a conversation's unrestricted nature. Ward and Vega [10] applied prosody to condition word probabilities in a conversational speech domain. In this study they find that using immediate prosodic context with a baseline 3-gram model achieves a 2.6% reduction in perplexity. In an extension to that same model, combining temporal information reduces perplexity down 8% from baseline [13]. This

temporal-prosodic model, when used in an ASR system, achieved a significant 1.0% reduction in WER (0.4% absolute) when applied to a German dialog corpus. These promising results using only immediate prosodic context led Ward and Vega [12] to use more prosodic features to potentially increase the information given to the LM. Within a six-second window centered at word onset, a feature set including both speaker and interlocutor volume, pitch and speaking rate features was calculated. Principal Component Analysis (PCA) was then applied to produce a feature set composed of 76 PCs. Using the top-25 best performing PC models achieved a 26.8% perplexity reduction. This model though makes the assumption that all features are equally informative. This naive assumption may actually hurt the overall model, by including features that could not be beneficial to the prediction task in any way.

2.3 Summary

Evidence from previous research suggests that prosody is a good source of information for the word prediction. Prosodic features are readily available for calculation from the voice signal, easily adaptable for use as context, and proven to improve ASR performance. While state of the art models achieve great perplexity reductions, they may suffer from the inclusion of features that introduce non-sufficient or noisy information to the language model. In this thesis, I evaluate the information contribution of prosodic features, choose the best, and show that a model that incorporates these features performs better than a model incorporating features in an ad hoc manner.

Chapter 3

Methodology

In this chapter, I explain the methodology to achieve the goals set forth in the previous chapter. The methods for calculating and evaluating prosodic features, the technique for combination with a n-gram LM, and the application domain are discussed.

3.1 Prosodic Features

3.1.1 Feature Calculation

While the main point of this study is to find the set of most informative features for use in a LM, the target goal for using this information in the first place remains in line with the goals set out by Ward and Vega [10]. Prosody here is used as a step towards exploiting cognitive state information for language modeling. Thus, the features calculated are direct [6] in the sense that they do not correspond to hand-labeled data. They are also not syllable-aligned, syllable-normalized, or computed over complete utterance. Rather they are calculated at 10ms intervals over fixed-size context windows.

The features chosen for this study are defined in Table 3.1. The features are derived from a basic set of features defined in [13]. The features were chosen out of convenience as

Respond, an in-house feature extractor, computes these features. These features also let me make a direct comparison between this study's model and the one built in [12]. Volume is chosen as an indicator of speaker engagement and dominance. This feature captures lexical stress patterns as well as patterns where loudness is used as a means of communicating opinion and a stance on a certain topic. Pitch features are strongly associated with a speaker's involvement in the conversation. Places where the speaker is laughing or uttering emotionally colored words are identified by this kind of feature. Speaking rate is able to identify speaker preparation and confidence. Words after slow speaking rates are found in areas of speech where the speaker utters fillers as a way to prepare their utterance. Fast speaking rate contexts characterize areas of speech where high-content words are uttered, namely place names and numbers [13].

Table 3.1: Prosodic Feature Definition

Prosodic Feature	**Context Window Size**	**Significance in Dialogue**
Volume	50 ms	Engagement and dominance
Pitch Height	150 ms	Involvement and lexical access
Pitch Range	225 ms	—
Speaking Rate	325 ms	Degree of preparation/confidence

Context window sizes were found with a hill-climbing approach for each feature in isolation, seeking to optimize perplexity for a LM incorporating this information. One thing to note are the rather small context windows over which perplexity was minimized. This

suggests that, for the prediction task, a prosodic features calculated over small windows of context contain relevant information for the prediction task. This is especially true for volume, the feature with the shortest high-value context window and the largest perplexity reduction when used in isolation with a LM [13].

After the calculation of the prosodic features, statistics for word occurrences preceded by a given prosodic context are generated. First, the prosodic features are "binned" from a continuous scale into four discrete categories (low, low-mid, mid-high and high). However, the thresholds for binning these features are decided in a different fashion than the binning process defined in [13]. The thresholds for the volume and speaking rate features are taken directly from the value distribution, using the quartile values as thresholds, effectively dividing the distribution of values into four regions. For the pitch features though, an extra step needs to be taken. The absence of pitch at different points throughout a conversation makes it more difficult to base thresholds solely on quartile boundaries. As missing pitch points account for 40% of the data points used for training, they need to be handled differently. One approach would be to assign these values the average pitch range/height seen in the data, however this is not realistic as pitch would be assigned to regions where there might be no speech at all. Setting those invalid points to zero pitch values would skew the data so badly that most contexts would be binned to the "high" categories, so that is not a great approach either. Instead, missing pitch frames are directly binned to the none category and are not taken into account for finding the quartile points of the pitch height/range value distributions.

3.1.2 Using Non-adjacent Features

In Ward and Vega [13] only immediate context information was used to condition word probabilities. Ward and Vega [12] followed that up by using information from past/future non-adjacent features in the PCA mixture. Here, non-adjacent features are defined as features that are offset by one or more feature window sizes from a given time point. Although words were not conditioned directly on the non-adjacent context in their model, this information contributed to the probability estimates of a word at different contexts. Here however, there is a need to condition words on these non-adjacent features to test the informativeness of a feature. Thus, I introduce offset prosodic models. Figure 3.1 illustrates these features.

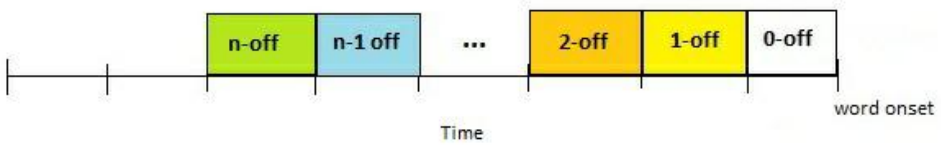

Figure 3.1: Offset prosodic features.

The models from previous studies can be thought of as 0-offset models, where the feature contributing information is the feature at the immediate context. Thus, n-offset models can be used to test the information contribution of features that originate at times offset from word onset.

3.1.3 Using Longer Context Features

One aspect of the PC model mentioned earlier that is worthy of study is the use of longer window sizes for features further away from word onset. In that model, features further away from word onset were aggregated into larger windows under the assumption that features further away from onset contain less information than windows closer to the point of interest. Thus, aggregating adjacent features further from point-of-interest (e.g., word-onset or word-end) could increase their contribution to a language model.

The method for increasing feature calculation is illustrated in Figure 3.2. The feature calculation is done by producing features over increasing window sizes. This is done by using Respond, an in-house prosodic feature extraction tool used for calculating the raw prosodic features defined in section 3.1.1. To achieve the calculation on increasing window sizes, features are calculated over context window sizes that grow in 50 ms window size increments. Minimum and maximum window sizes are set to 100 ms and 500 ms respectively. To create an overlap between windows, as shown in Figure 3.2, windows are referenced in 50 ms window size offsets over the context limit. The context limit is defined as three seconds from point-of-interest, matching the context used in [12]. The context space is taken from past context, features calculated on context previous to word onset, and future context, features calculated on context situated at times future to word-end time. Features are incorporated from both speaker and interlocutor tracks. Taking features over this space results in a total feature space of 7104 prosodic features.

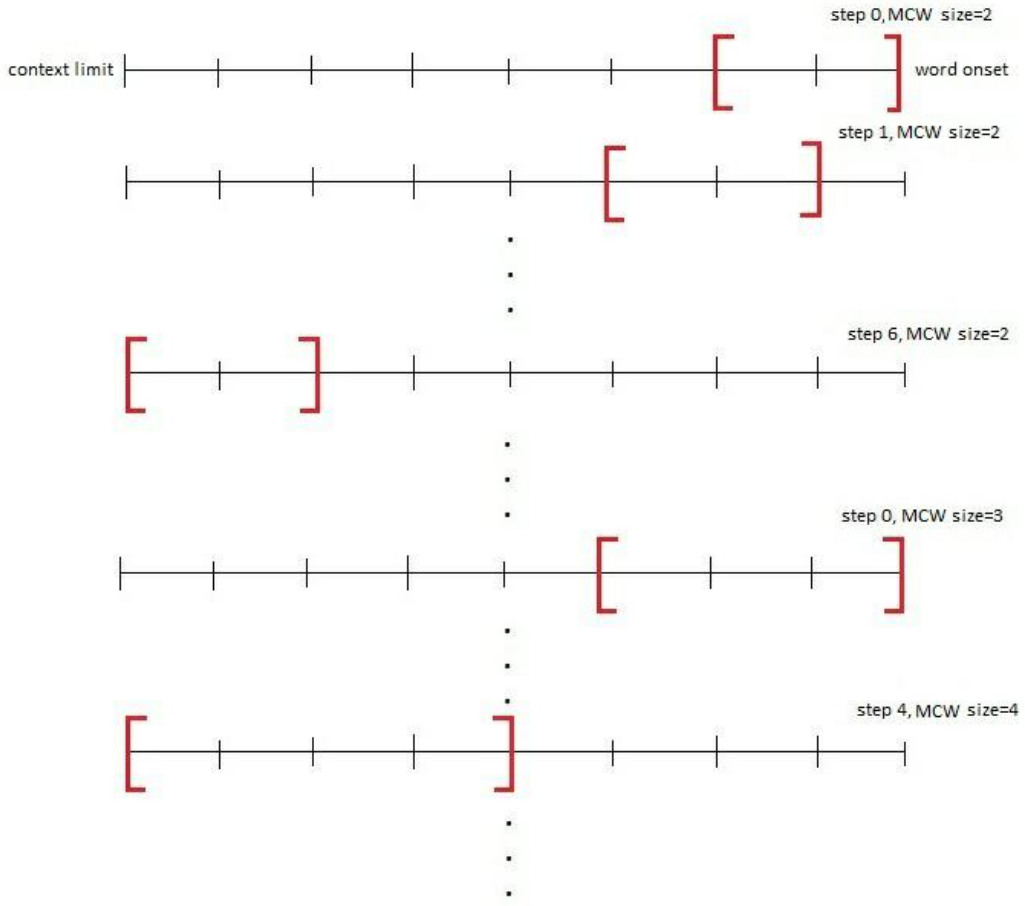

Figure 3.2: Feature calculation over increasing window sizes.

3.2 Use in a Language Model

The method used to combine prosodic information computed with a LM is the same used in [9]. This method is summarized in this section.

3.2.1 Combination with N-grams

Once binned, these discrete categories are used as context tokens to count the number of occurrences of a word after a given prosodic feature context. Thus, each word has a probability of occurring at a given prosodic context. For example, I can compute the probability of the word *the* happening after silent or high volume contexts, using the statistics over the training corpus.

These probabilities are then turned into probability ratios. These ratios are given by equation 3.1. For each word w_i at a given prosodic context x, I can compute the prosodic context probability, $P_{prsdy}(w@x)$. By taking the ratio of the prosodic context probability to the word's overall probability in the corpus, $P_{uni}(w_i)$, an estimate of the probability boost given by prosodic information is given. This ratio I call the R-Ratio. With an R-Ratio ratio greater than 1 at a given context, a word is more probable. With an R-Ratio less than one, a word is less probable at that context.

$$\mathrm{R}(\mathrm{w}_i@x) = \frac{P_{prsdy}(w@x)}{P_{uni}(w_i)} \quad (3.1)$$

However, some problems do arise from this approach. One of those problems is data sparsity. There are words that do not happen at every prosodic contexts, creating zero counts when generating the word counts, ultimately creating zero probabilities. To handle this, I use add-1 smoothing for these words. Another problem is the lack of information seen

from words that are too infrequent at certain contexts. To avoid this, the informativeness of a word's R-Ratio is estimated using the χ^2 test. From this test, the p-value of this hypothesis, p, is computed, and from that the confidence in the hypothesis, q=1-p, is computed. Finally, the R-Ratio is raised to the *qth* power, resulting in the S-Ratio, given by equation 3.2.

$$\mathrm{S}(\mathrm{w}_i@x) = R(w_i@x)^q \quad (3.2)$$

These S-Ratios are then used as scaling factors to the n-gram probability, shown in equation 3.3. Across the whole vocabulary, each scaling factor is associated with a parameter k that increases or decreases that given feature's contribution to the overall combination. This paramater is fixed for each feature. The scaled values are then normalized to ensure true probability estimates. Notice that this approach doesn't limit the amount of information that can be applied: the information for multiple prosodic features can be combined into the LM in the form of multiple scaling factors.

$$\mathrm{P}(\mathrm{w}_i@x|c) = P_{lm}(w_i|c) * S(w_i@x)^k \quad (3.3)$$

3.3 Evaluation Scheme

3.3.1 Corpus

As pointed out earlier, this model is applied to the word recognition task for conversational speech. The Switchboard corpus is used for LM evaluation. This corpus is a collection of short telephone conversations on light topics (e.g. movies, music, and light politics) between mostly unacquainted adults [3]. I retrieve relevant prosodic features using the ISIP transcriptions of the Switchboard corpus [4]. These transcriptions are time-aligned at the word level, allowing me to retrieve prosodic feature at or around the points of interest for each word.

From this corpus, 981 tracks, consisting of about 80 hours of speech and 650,000 words are used as training for both the baseline LM and the prosodic models. A held out set of tracks consisting of 35,000 words was used as tuning data to find the optimal set of meta-parameters, the most important being the k exponents associated with each scaling factor. Final evaluation is done on a separate set of data. This test set consists of 45 tracks from Switchboard, containing 28,000 words and making up 4 hours of dialog.

3.3.2 Performance evaluation

To evaluate the performance of each prosodic model, perplexity is the measure of choice. Perplexity, in layman's terms, represents the difficulty of recognizing the current word. Thus, a lower perplexity value is better. The baseline LM is the a back-off trigram model

implementation from the SRI Language Modeling Toolkit [7]. Vocabulary for this LM is limited to 5000 words, with other words treated as unknown tokens. Baseline perplexity for the test set was 109.449. Feature selection is based solely on perplexity reduction, as will be discussed below.

Chapter 4

Initial Observations

The S-Ratios not only allow for relatively easy integration for information in a LM, they also enable a quick way to observe how words behave depending on prosodic context. If conditioning on prosodic context gave no relevant information, then all ratios generated would be 1.0, basically saying that prosodic information is no better than using unigram probabilities.

The probability ratios discussed in this chapter correspond to past speaker context features for the most frequent words seen in the corpus.

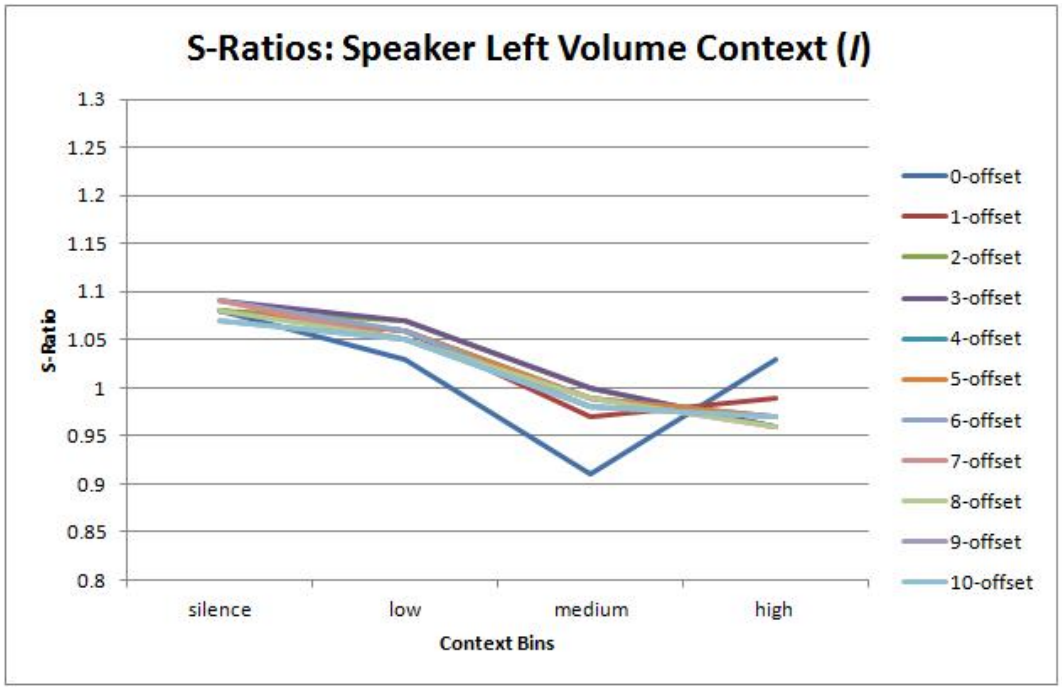

Figure 4.1: S-Ratios for word *I*.

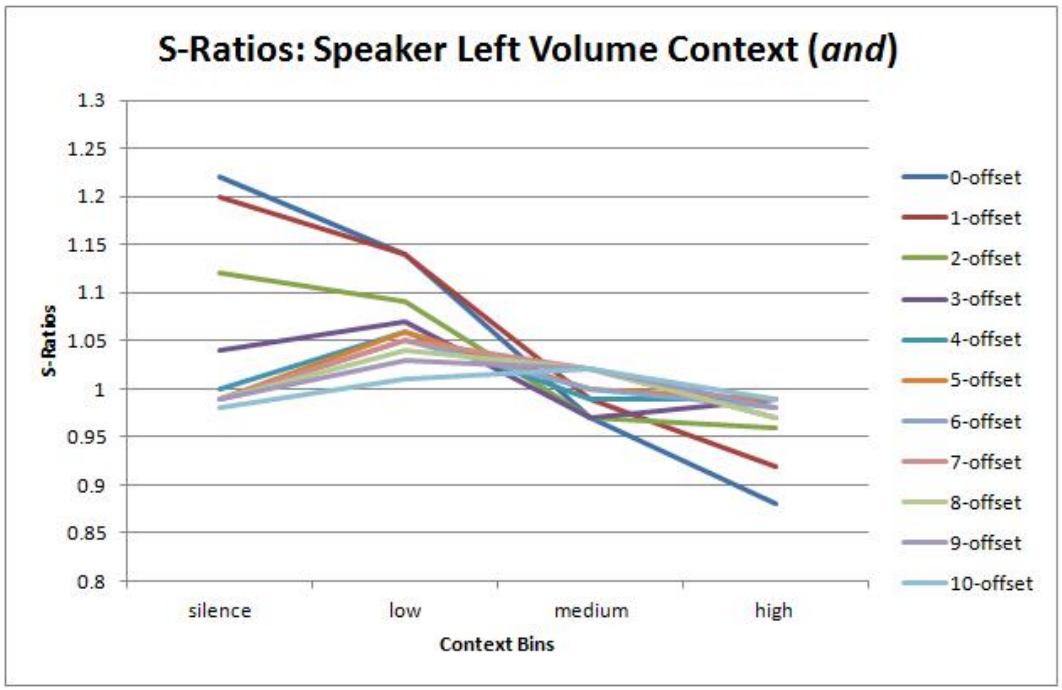

Figure 4.2: S-Ratios for word *and*.

Figure 4.1 shows the S-Ratios for the word *I* when conditioned on the speaker's left volume context. Conditioning on immediate context (0-offset) yields the most variation. Here the word *I* is more common after none (e.g., silence-level volume), low, and high volume contexts when compared to the word's overall probability in the corpus. However, the model indicates that the word is uncommon at medium volume contexts, where the ratio falls below 1.0. For offsets preceding immediate context, the trend for the ratios remains similar, almost converging to the same behavior, where S-Ratio values don't change much between offset windows. For the word *and*, shown in Figure 4.2, there is a similar pattern.

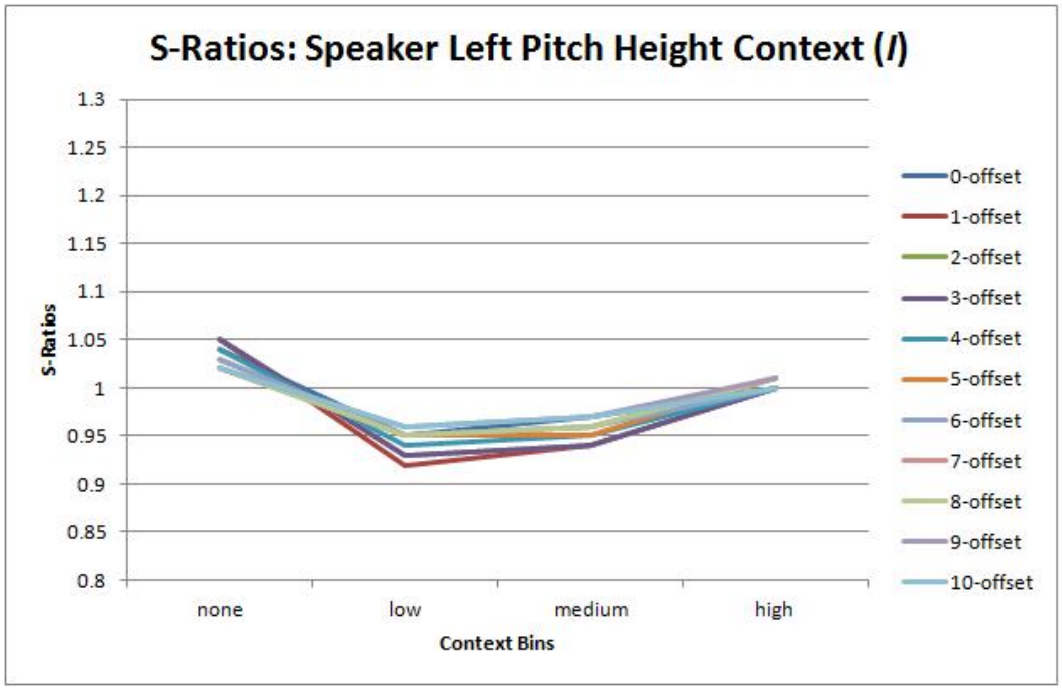

Figure 4.3: S-Ratios for left pitch height context.

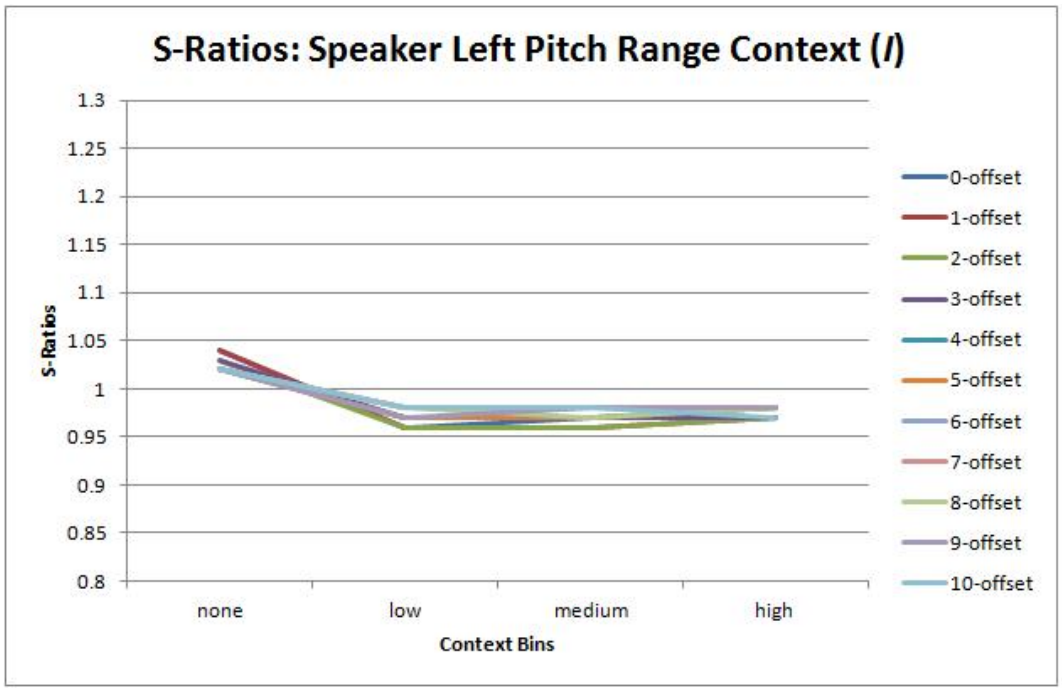

Figure 4.4: S-Ratios for left pitch range context.

Here, the pattern seen for the word I, where ratios stop changing after a certain offset, is more pronounced. After the third offset window, S-Ratio values change only slightly basically keeping the same behavior as they move further away from word onset. Looking at the pitch features (Figures 4.3 and 4.4) and speaking rate (Figure 4.5) reveals similar patterns.

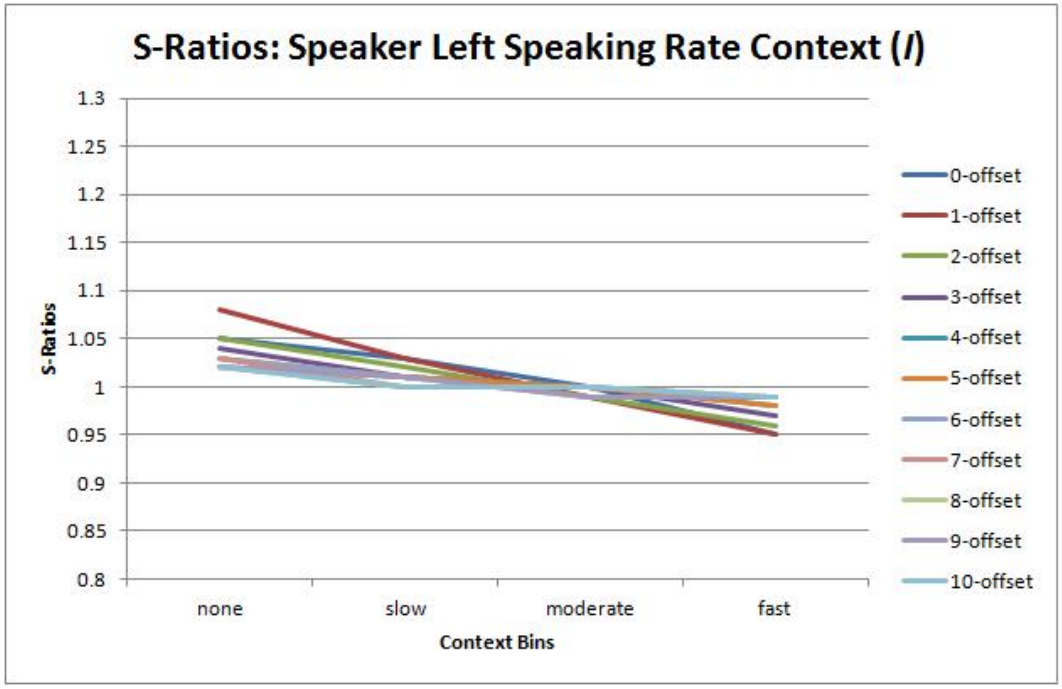

Figure 4.5: S-Ratios for left speaking rate context.

Pitch height/range features also exhibit the same offset limit, after which S-Ratios don't change. This trend is an interesting one though. If this property holds for other words as well, then it suggests that there is an offset point for these features where conditioning on farther contexts yields little additional information than the closer information. To see

if this notion holds or not, the S-Ratios for all words in the vocabulary were analyzed. Mean absolute difference was used as the measure of change between S-ratios. Figure 4.6 summarizes the results.

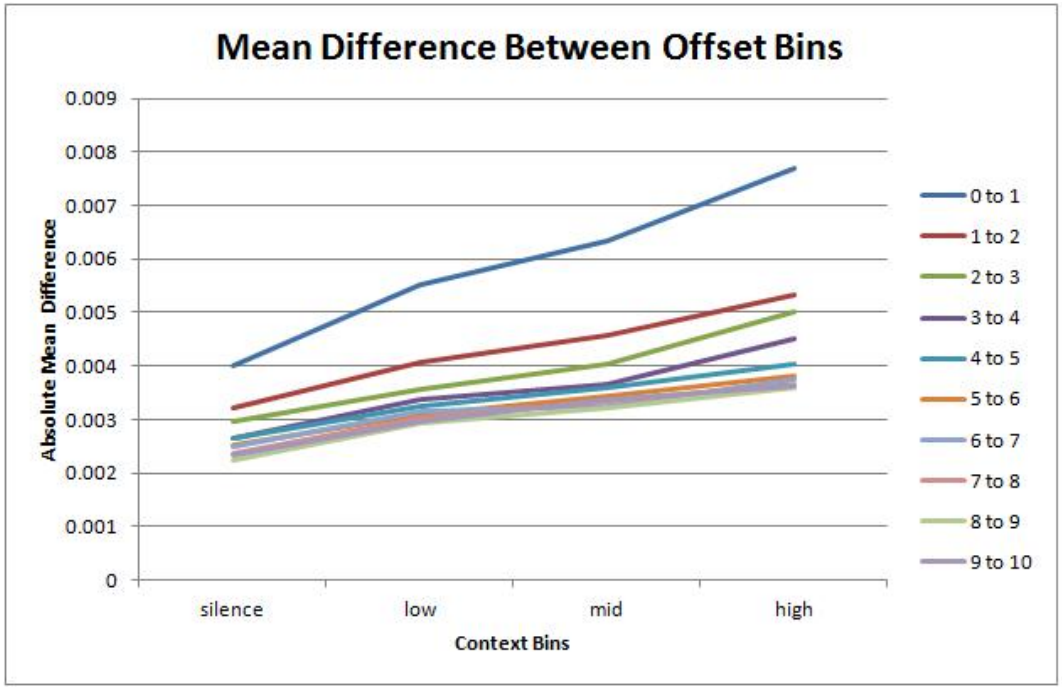

Figure 4.6: Offset S-Ratio Mean Absolute Difference

The graph shows the mean absolute difference between offset S-Ratios for all corresponding words in the vocabulary when conditioning on volume. Differences from zero to one offsets are larger compared to all differences between other offsets, showing that S-Ratio values between those offsets do change. However, moving away from zero offset those differences decrease. Once past the third offset window, differences become smaller. When reaching the fifth or sixth offsets, differences, although not zero, are virtually the same, indicating that conditioning on farther context is almost as beneficial as conditioning

on nearer non-adjacent features.

Chapter 5

Feature Evaluation

In this chapter, the results of conditioning word probabilities on prosodic features in the context space is presented. I present an analysis for the information contribution of these features to the word prediction task.

5.1 Non-adjacent Feature Models

As discussed in section 3.1.2, I calculate scaling factors by conditioning word counts on non-adjacent offset features for both past and future context, and both on speaker and interlocutor context. Models are generated and evaluated for each non-adjacent feature that contained in the defined context limit. Models are evaluated in isolation on the tuning set.

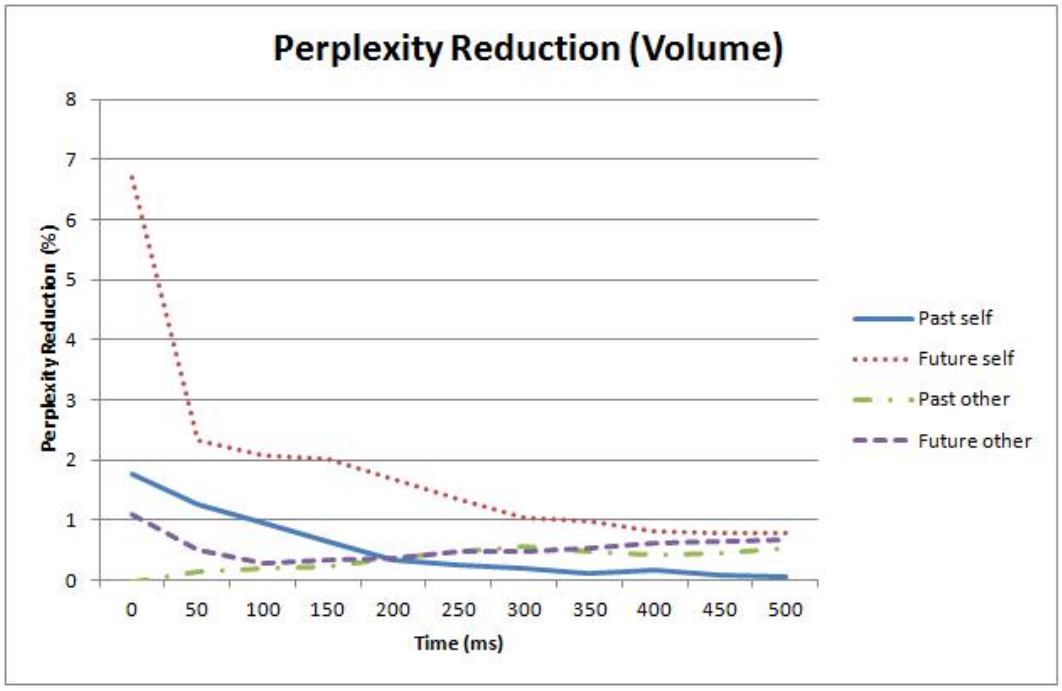

Figure 5.1: Perplexity reduction for non-adjacent volume models.

Figure 5.1 shows perplexity reductions relative the baseline trigram model when using volume information for each of the first 10 non-adjacent feature windows. Previous studies [13] show that same-speaker volume features provide the largest perplexity benefit over baseline. The same thing is seen here when using non-adjacent windows close to the point-of-interest, both for past and future context. Future speaker context in particular seems to provide largest amount of benefit as the benefit given at further offsets constantly remains above other features at the same time offset. While future speaker context exhibits this behavior, past speaker context does not. Past speaker context only performs well for the first four offset for the first 200 ms. After that point, perplexity benefits degrade below

the 0.3% reduction point, producing benefits well below future context and below those produced by interlocutor features. Interlocutor volume features from past contexts provide small benefit at windows close to the point-of-interest. The real benefit from these features comes at offset windows farther from the point-of-interest, as seen in the figure. This growth may indicate that there is a period of time between interlocutor features and speaker words that needs to pass for that feature to be useful enough for the prediction task.

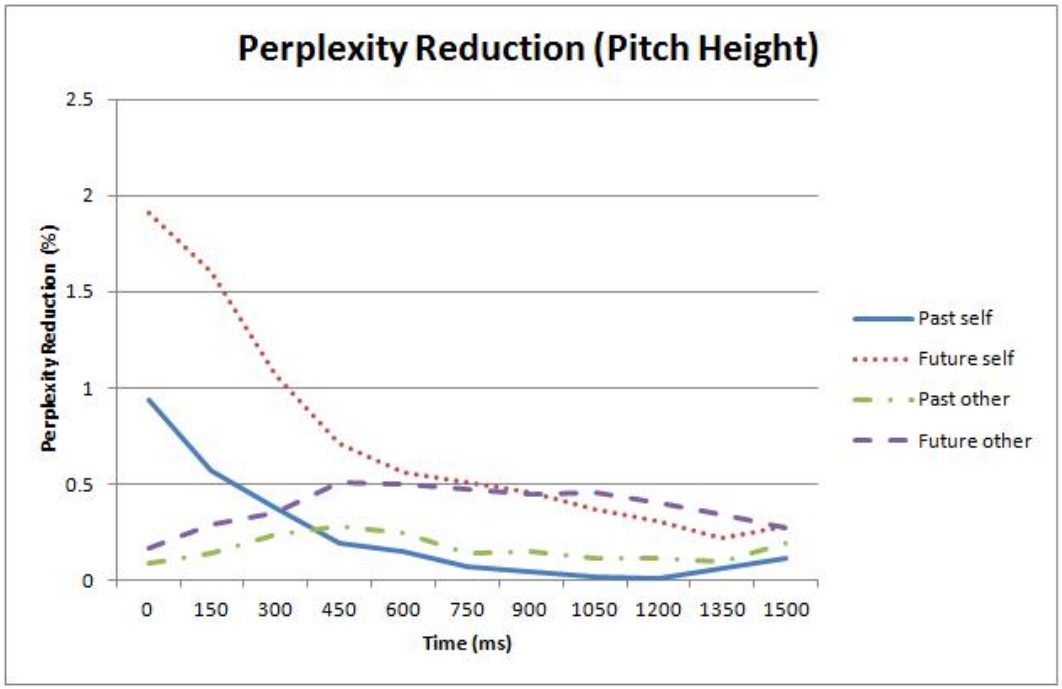

Figure 5.2: Perplexity reduction for non-adjacent pitch height models

For pitch height features, a similar pattern is observed. This is shown in Figure 5.2. Future speaker features are dominant close to the point-of-interest as compared to other features. Unlike volume features though, the contribution from interlocutor future context

becomes large enough to overtake future speaker pitch height past 900ms from the point-of-interest. Past speaker pitch height readings are still informative at immediate points, but that contribution degrades below the 0.5% level after approximately 300ms and below 0.1% after 750ms from the point-of-interest. Past interlocutor information performs below the 0.5% reduction level consistantly along all offsets.

Pitch range features, whose perplexity reductions are illustrated in Figure 5.3, show similar trends. Although future speaker features have low perplexity reduction immediately at word end, these features still produce higher reductions within the first 500ms from the point-of-interest than to the other features. speaker past features are most informative immediately before the point-of-interest, becoming almost null at 900ms. After that point though, benefits fall to the 0.2% level past 2 seconds from word onset. Interlocutor features show most information contribution at 450ms and 675ms for past and future features respectively.

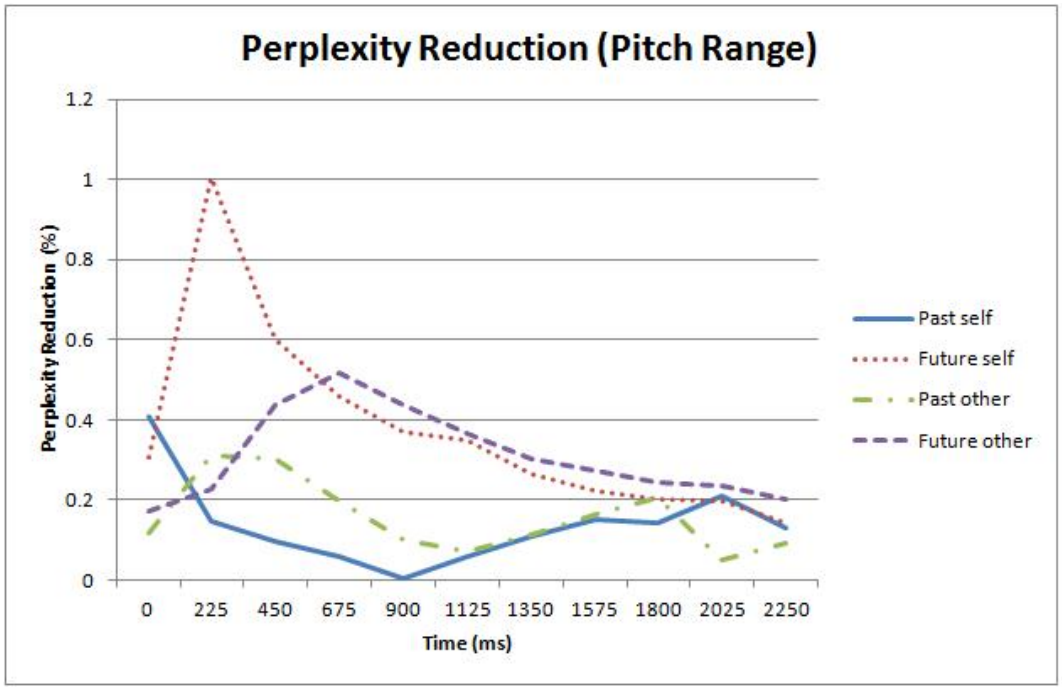

Figure 5.3: Perplexity reduction for non-adjacent pitch range models

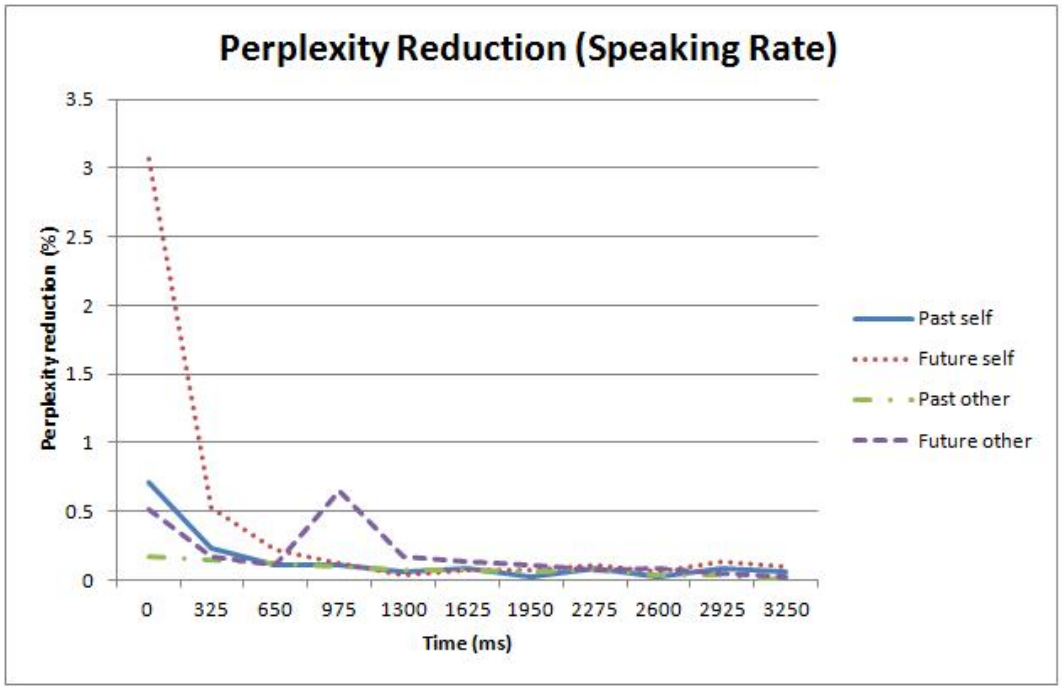

Figure 5.4: Perplexity reduction for non-adjacent speaking rate models

speaker future speaking rate features, shown in Figure 5.4, exhibit the second biggest perplexity reduction of all features at 3.07%, then dropping down to 0.53% at 650ms from the point-of-interest. Future interlocutor features show different behavior from that in the previous features. Future interlocutor features are informative at the point-of-interest (0.5% perplexity reduction) and then drop below 0.2% only to rise up to 0.65% at 975ms from the point-of-interest.

5.2 Longer-Width Window Feature Models

As with the non-adjacent features, S-Ratios for features computed over longer window sizes are calculated for both speakers conditioning on both past and future context. In this section, I present the results of applying this information to the word prediction task, as a way to judge the utility of the various possible features.

5.2.1 Past speaker features

In this section, I describe the perplexity reductions obtained by using longer-width speaker features.

Figure 5.5 shows perplexity reductions for past volume features over a three second context limit. As also seen earlier, volume features are particulary strong close to word onset, achieving up to 1.58% perplexity reduction for volume calculated over a 100 ms window. As the window width increases, perplexity reductions close to the point-of-interest decrease, suggesting that small, fine grained windows are suitable for contexts close to the point-of-interest. However, these perplexity reductions fall fast with time. Reductions only manage to stay above the 0.5% line for about 150 ms from word onset, and they are below 0.2% reduction for earlier contexts as seen in the figure. While there are regions where larger windows are more informative than smaller windows, the differences between these perplexity benefits is small and may not be significant.

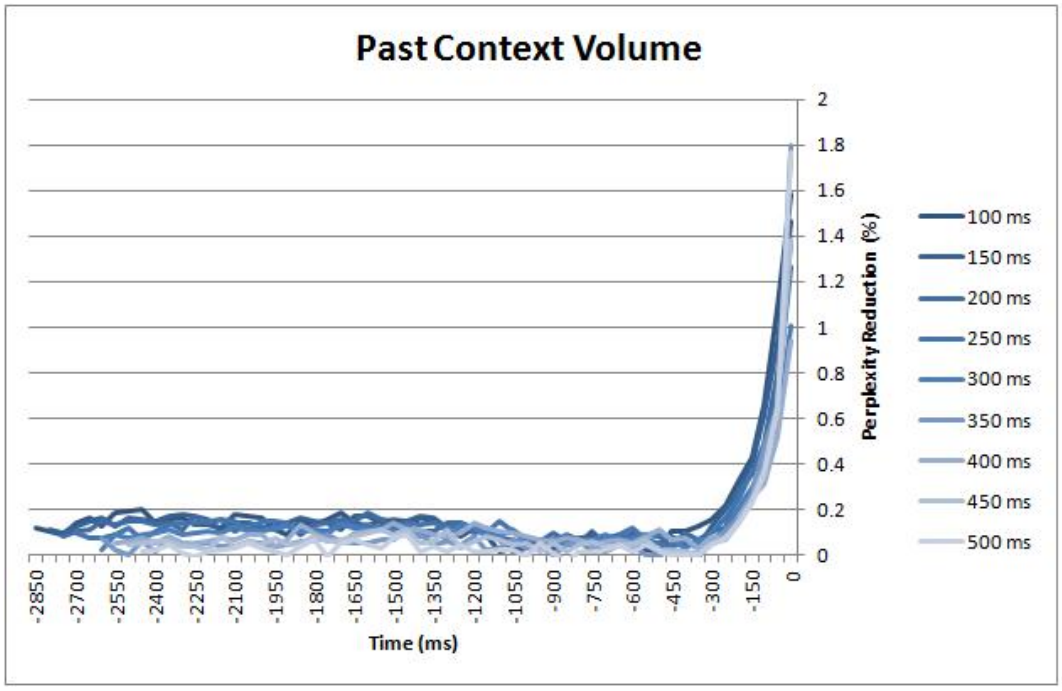

Figure 5.5: Perplexity reduction for past volume models

Pitch height features, whose reductions are shown in Figure 5.6, exhibit similar patterns. Although the reduction benefits over 0.5% last longer than they do for volume features, the reductions still fall below 0.2% past 250 ms from word onset for all width lengths. Pitch range features though (Figure 5.7, exhibit other characteristics. Both for windows from the point-of-interest and on more distant contexts, larger windows are evidently more informative than smaller widhts. Smaller width features actually hurt the model at contexts from 250 ms to 1300 ms, reaching perplexities as bad as 0.11% worse than baseline. When the window width exceeds 200 ms, these features perform better than baseline consistently within the three second context limit.

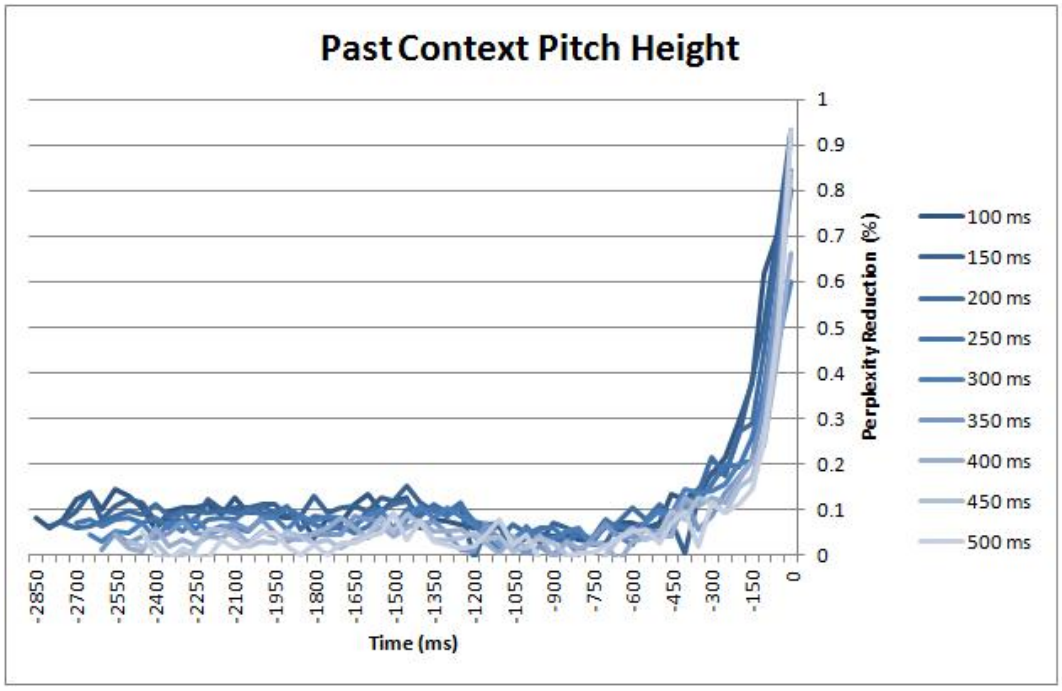

Figure 5.6: Perplexity reduction for past pitch height models

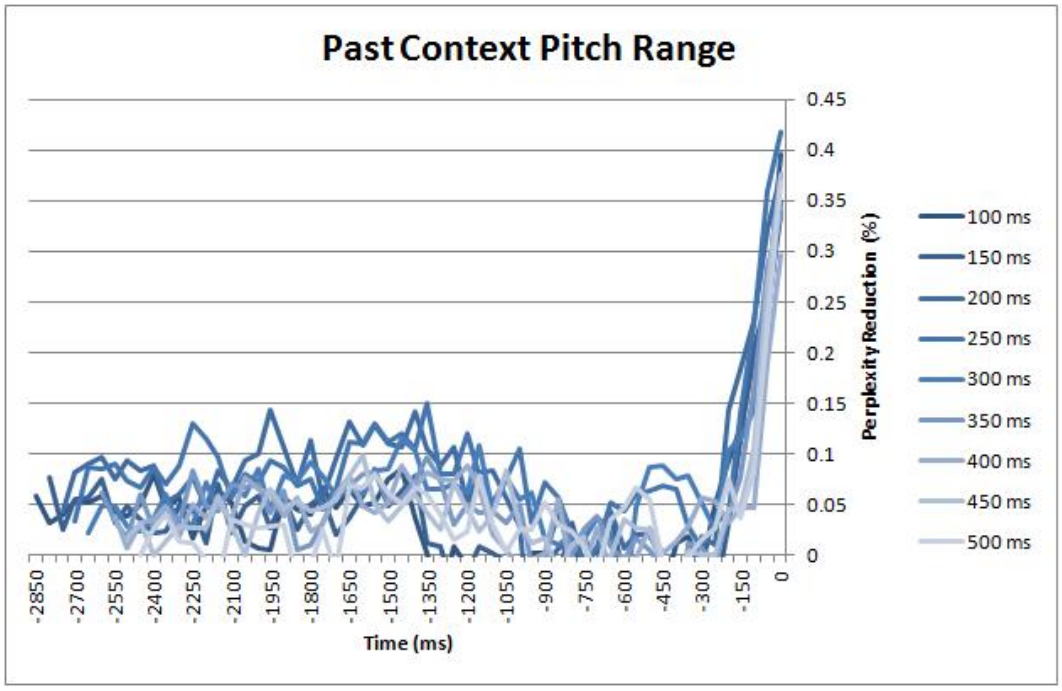

Figure 5.7: Perplexity reduction for past pitch range models

Speaking rate features, Figure 5.8, exhibit similar behavior to volume features, where small width features provide better benefits up to 150 ms from word onset. After that point, reduction stay constantly below 0.2%.

From these and previously seen trends, a pattern starts to emerge. As previously drawn from the behavior of S-Ratios in Chapter 4, information from past context features seems to be strong only for a small amount of time before word onset. This is clearly seen in the perplexity benefits produced over time. The best benefits are seen for features close to the point-of-interest. As I condition word probabilities on earlier contexts, information contribution dwindles across all models.

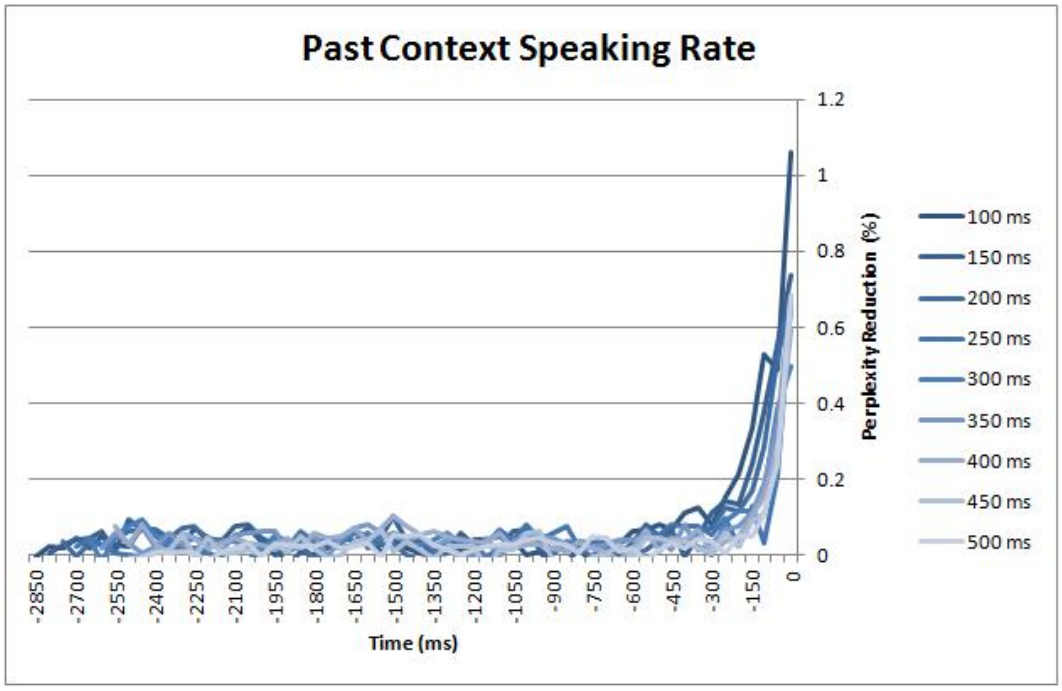

Figure 5.8: Perplexity reduction for past speaking rate models

5.2.2 Future speaker features

When conditioning on a speaker's future context, different patterns emerge.

Figure 5.9 shows perplexity reductions when conditioning on speaker's future volume context. Here small width contribute most information close to the point-of-interest. For future volume, perplexity benefits stay above 0.5% for longer. For feature widths of 100-350 ms, volume features produce benefits greater than 0.5% up to 300 ms from the point-of-interest. Past this point, perplexity reductions again dwindle down to smaller values. While this behavior is similar to the one seen for past volume features, there is an interesting region where, for some window sizes, perplexity benefits increase to a local maximum and then

descend. This is seen for feature widths of 200-300 ms and 400 ms in the region 100-200 ms from the point-of-interest. For each of these features, perplexity benefits drop from their global maximum at the point-of-interest and rise for a brief period of time.

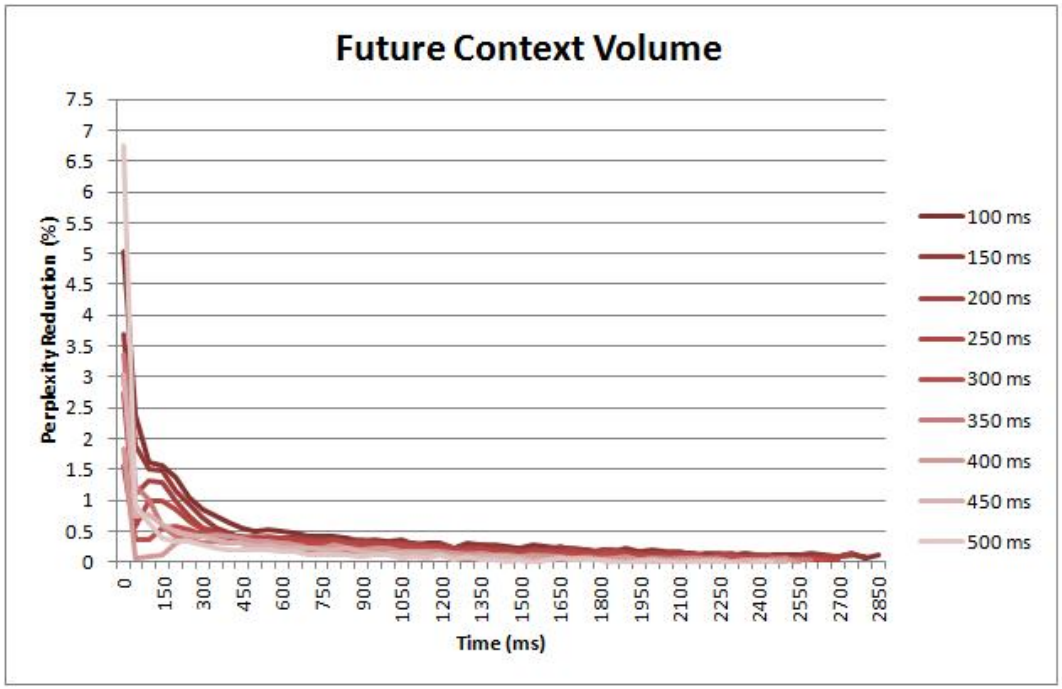

Figure 5.9: Perplexity reduction for future volume models

For future pitch-height features, the same rise in perplexity after a steep drop is seen. Figure 5.10 shows the perplexity benefits. Features close to the point-of-interest do better than those at farther points. Perplexity benefits drop below 0.5% past 250 ms of future context, becoming very small past 1.5 seconds. The behavior for feature widths of 300 and 400 ms is interesting in that there is a large region (250 ms for 400 ms wide features) where perplexity benefits reach a minimum and rise again after 300 ms. For future pitch range

features, shown in Figure 5.11, conditioning on features past 250 ms produces benefits constantly less than 0.5% perplexity reduction. Although even within the region of 0-250 ms from the point-of-interest, small widths do best in comparison to features calculated over larger windows.

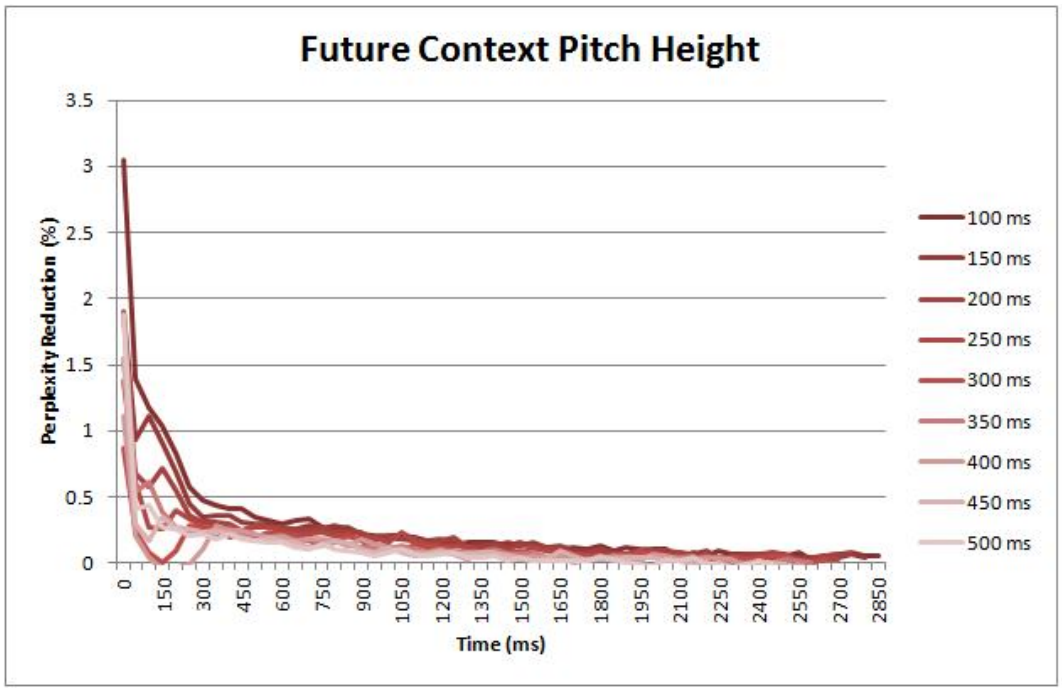

Figure 5.10: Perplexity reduction for future pitch height models

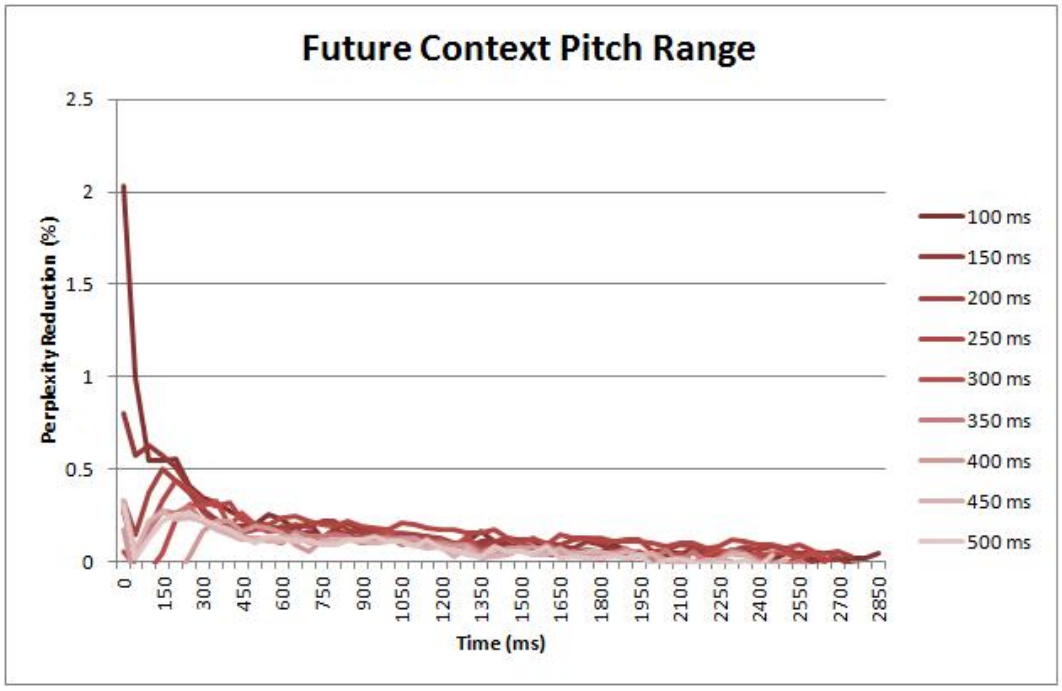

Figure 5.11: Perplexity reduction for future pitch range models

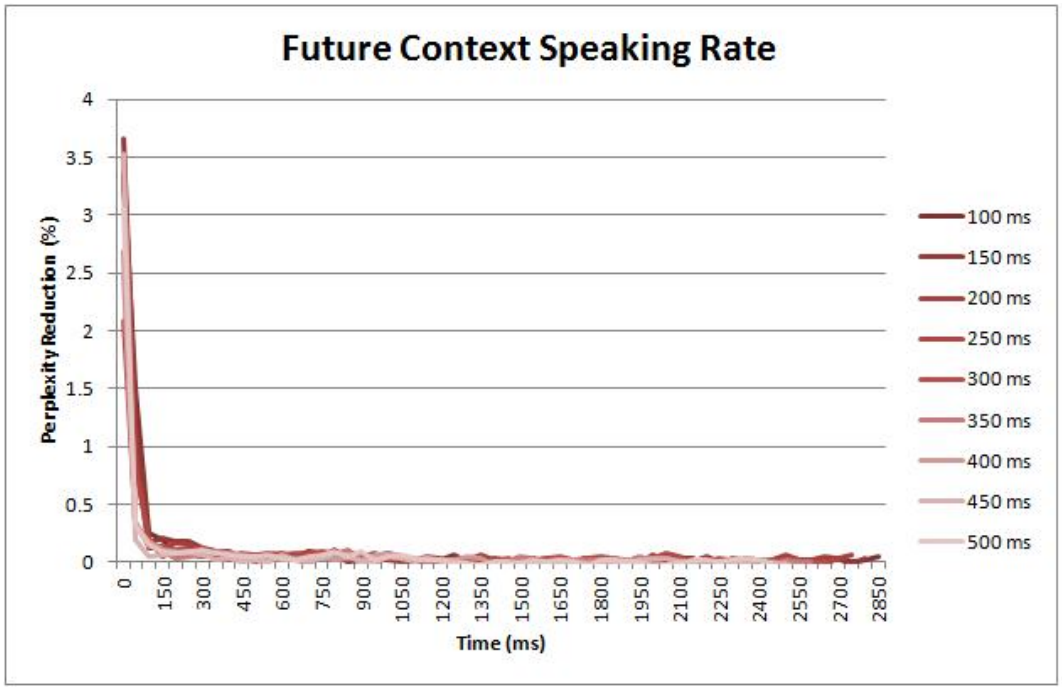

Figure 5.12: Perplexity reduction for future speaking rate models

Future speaking rate features (Figure 5.12) are only valuable very close to the point-of-interest. The steep drop in benefits after 100 ms makes it clear that using further rate context is not beneficial for the prediction task. Calculation over longer windows benefits perplexity reduction, as seen from the performance produced by calculating over windows of 350 ms and 450 ms. However this only lasts for the first 100 ms after the point-of-interest, after this point perplexity benefits stay close to a mere 0.1% reduction

The results shown from conditioning on speaker past and future features confirm one of this study's previous observations: Speaker features that are close to the point-of-interest are best suited for the word prediction task.

5.2.3 Past interlocutor features

Interlocutor features show different characteristics. First we consider past context features.

Figure 5.13 shows the perplexity benefits when conditioning on the interlocutor's past volume features. Here, volume features exhibit something interesting. Perplexity benefits peak away from the point-of-interest, unlike what was seen for speaker volume features. Most features, with the exception of those calculated over 400 ms, peak in the region of 250-350 ms from the point-of-interest. While these volume features produce benefits that are not large, short width features do best, reaching perplexity benefits of up to 0.75%. Features calculated over 400 ms peak earlier than shorter windows (approximately 150 ms) and provide benefits above 0.5% up to 350 ms from the point-of-interest. Pitch features show similar trends.

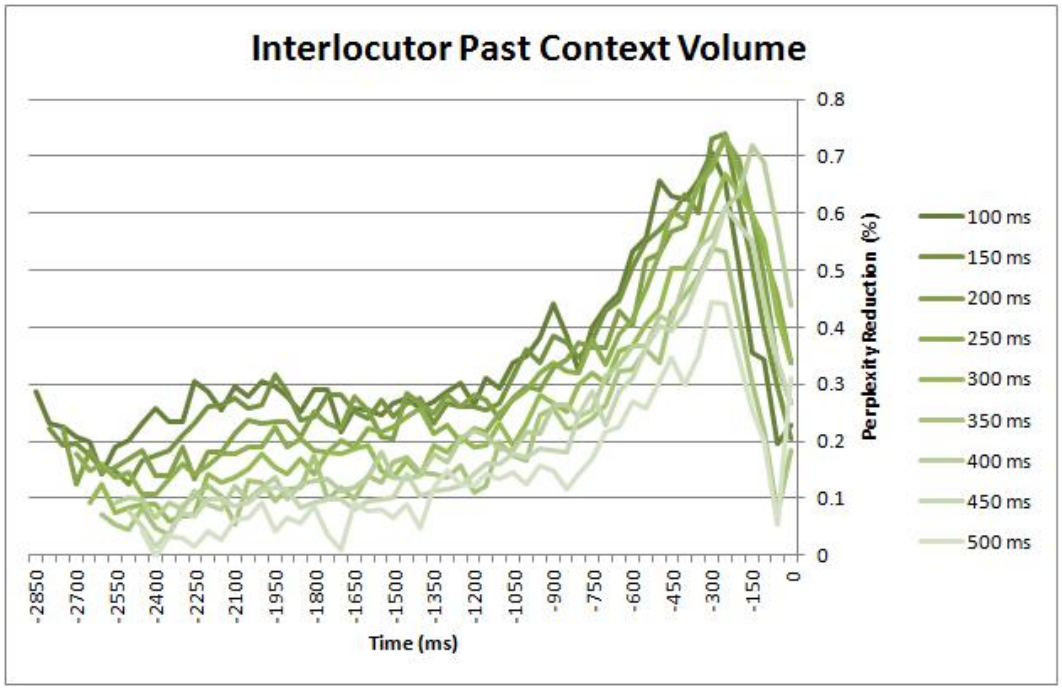

Figure 5.13: Perplexity reduction for interlocutor past volume models

Figure 5.14 illustrates the behavior of perplexity benefits when conditioning on interlocutor past pitch height features. Although benefits are not as big as those given by volume features, benefits also peak at points offset away from the point-of-interest. The peak for small width features happens in the region of 300-600 ms from the point-of-interest. Larger widht features peak early at approximately 200 ms from the point-of-interest for 350 ms wide features. After 600 ms, perplexity benefits start decreasing constantly until 1400 ms, where benefits from features 100 and 150 ms wide rise again. Pitch range features peak earlier than pitch height features. Shown in Figure 5.15, range features peak from 150-300 ms from the point-of-interest. Interestingly enough, range features calculated over 400 ms

do best, reaching 0.4% perplexity reduction at 150 ms from word onset. Features calculated over smaller windows (e.g. 100 and 150 ms) provide smaller benefits constantly throughout the three second context space, overtaking longer windows past the two-second mark.

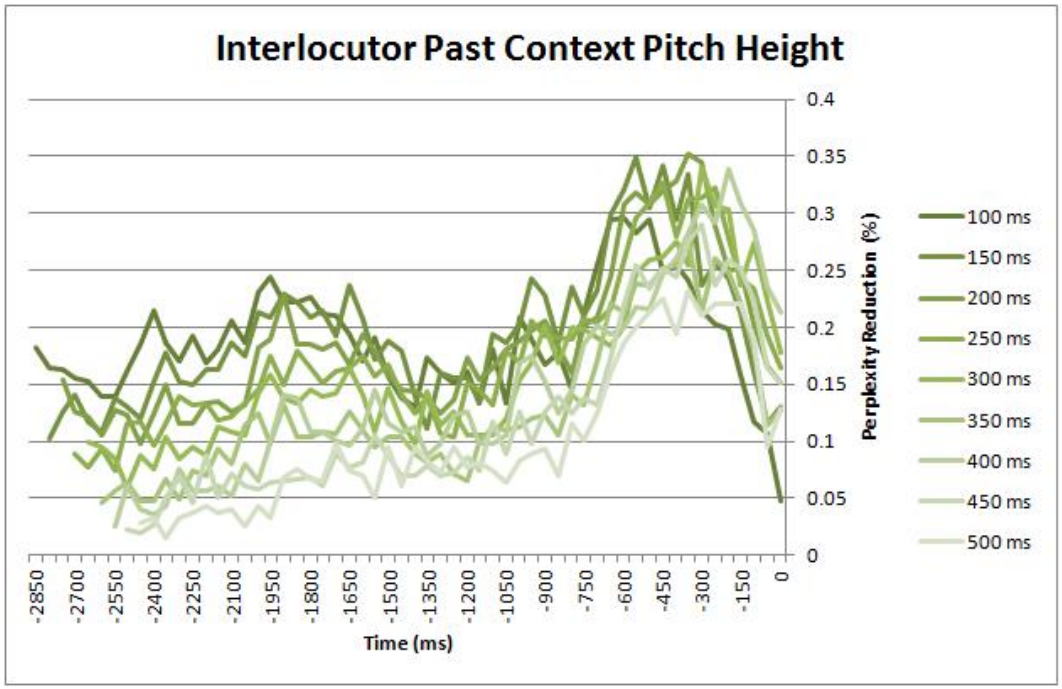

Figure 5.14: Perplexity reduction for interlocutor past pitch height models

Among interlocutor features, past speaking rate provides the lowest benefits for perplexity reduction. In contrast to the other features, as shown in figure 5.16, speaking rate features peak at the point on interest, only to decline constantly after that point.

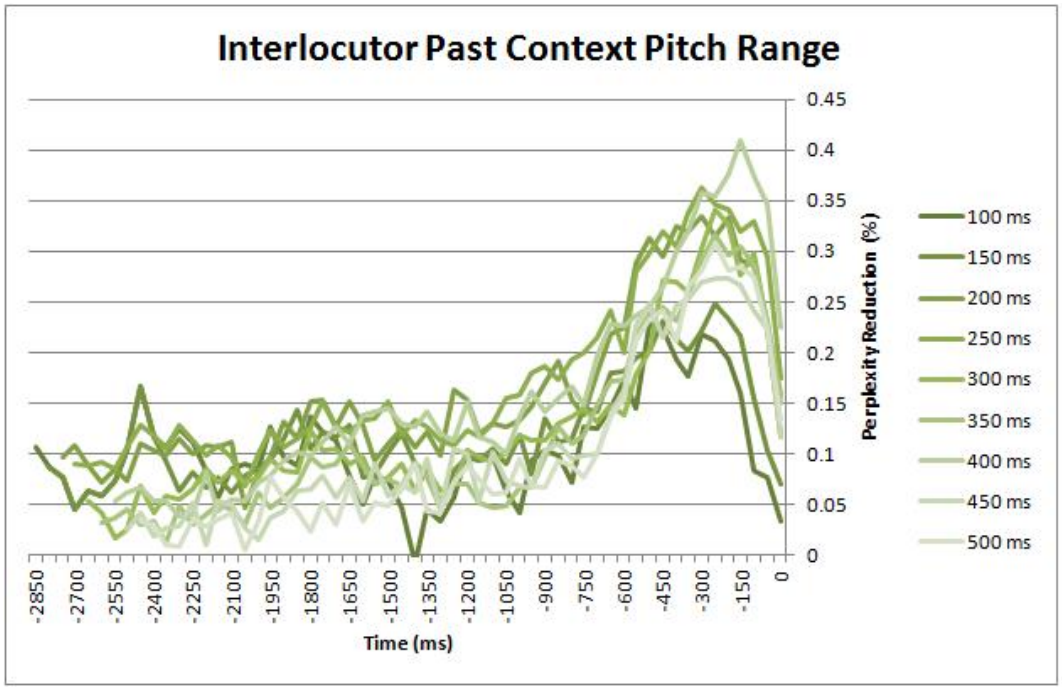

Figure 5.15: Perplexity reduction for interlocutor past pitch range models

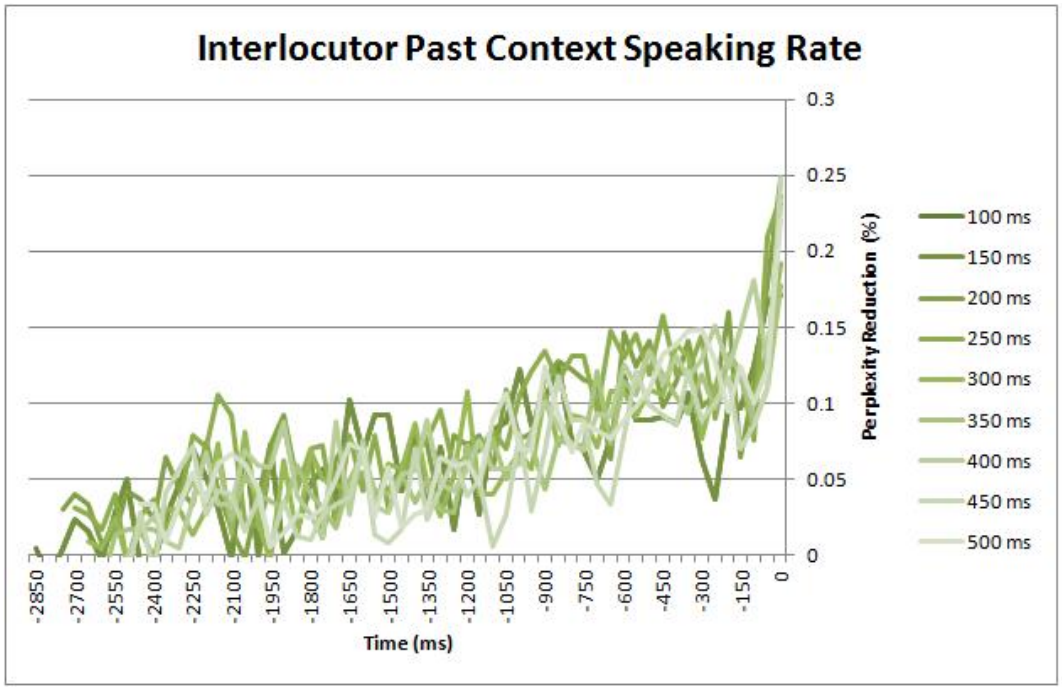

Figure 5.16: Perplexity reduction for interlocutor past speaking rate models

5.2.4 Future interlocutor features

Future interlocutor features exhibit different trends.

Volume features in particular show similar behavior to that of speaker volume features. This is depicted in Figure 5.17. Peaking at the point-of-interest, future interlocutor volume feature effects drop from their peak over a region of 50-100 ms from the point-of-interest. After this point, benefits rise again almost to the level where they peaked. This peak happens roughly at a region 350-650 ms from the point-of-interest. After this point, benefits begin to slowly fall again. Conditioning on features calculated over smaller windows (e.g., 100-200 ms) provides the best benefits across the whole three-second context window space.

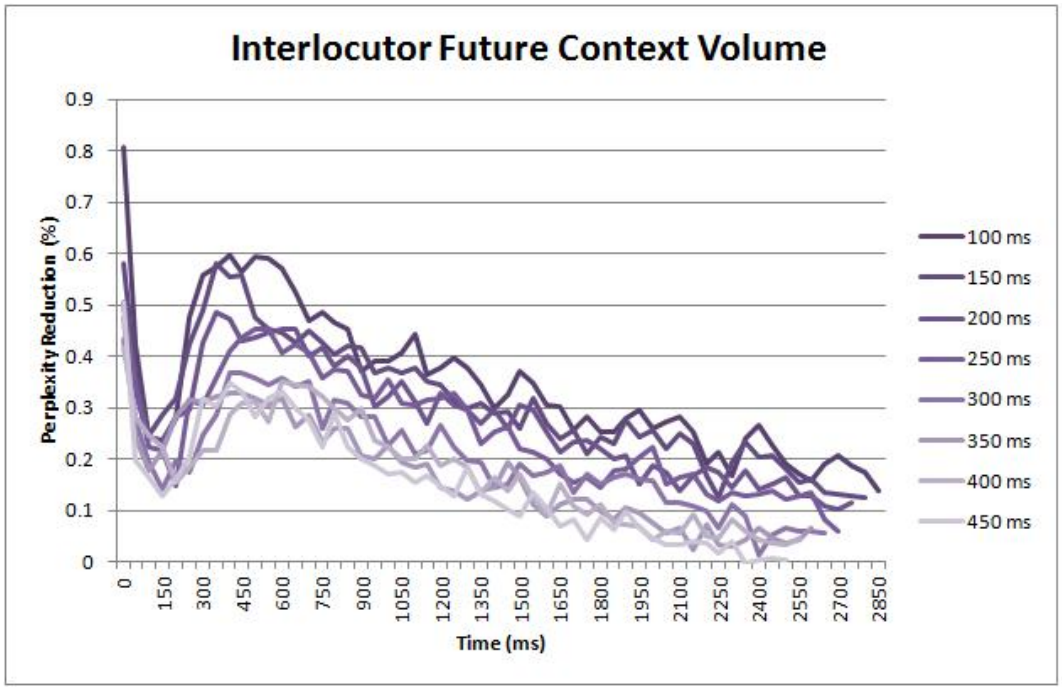

Figure 5.17: Perplexity reduction for interlocutor future volume models

Pitch features, whose benefits are shown in Figures 5.18 and 5.19, show similar behaviors to their past-context counterparts. Pitch height features peak approximately 400 ms from the point-of-interest and slowly decline after that. Pitch range features gain their peak later, at around 600 ms, pointing to the possibility that, for interlocutor context, the important information from these features appears at times that are not close to the point-of-interest. For future pitch contexts, calculating features over smaller windows seems best as they consistently provide better benefits over the whole feature space.

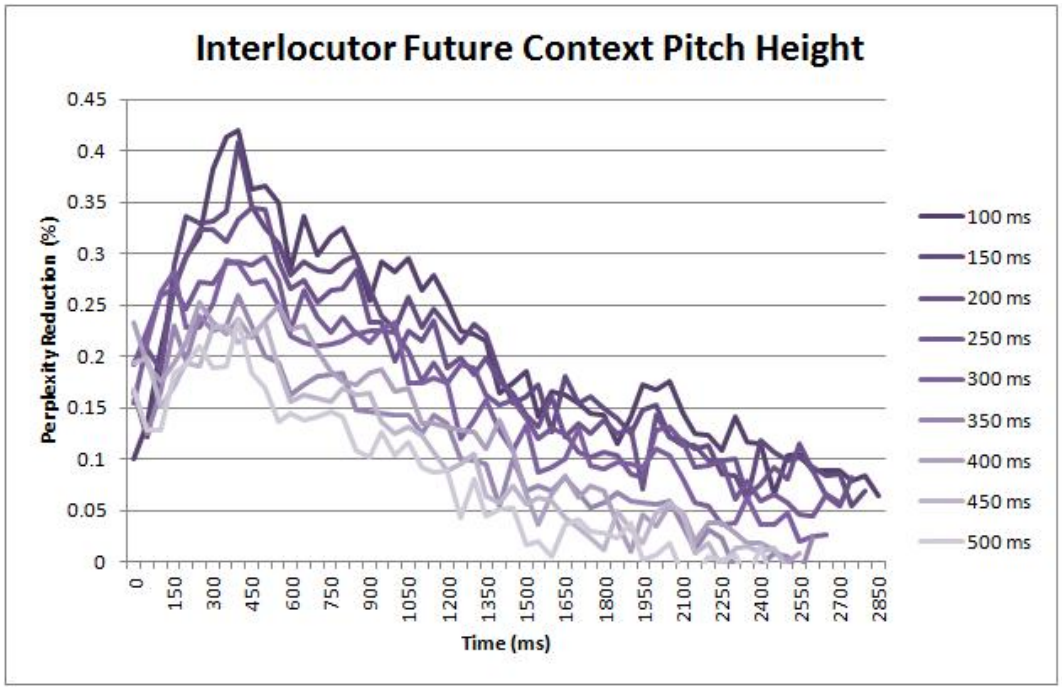

Figure 5.18: Perplexity reduction for interlocutor future pitch height models

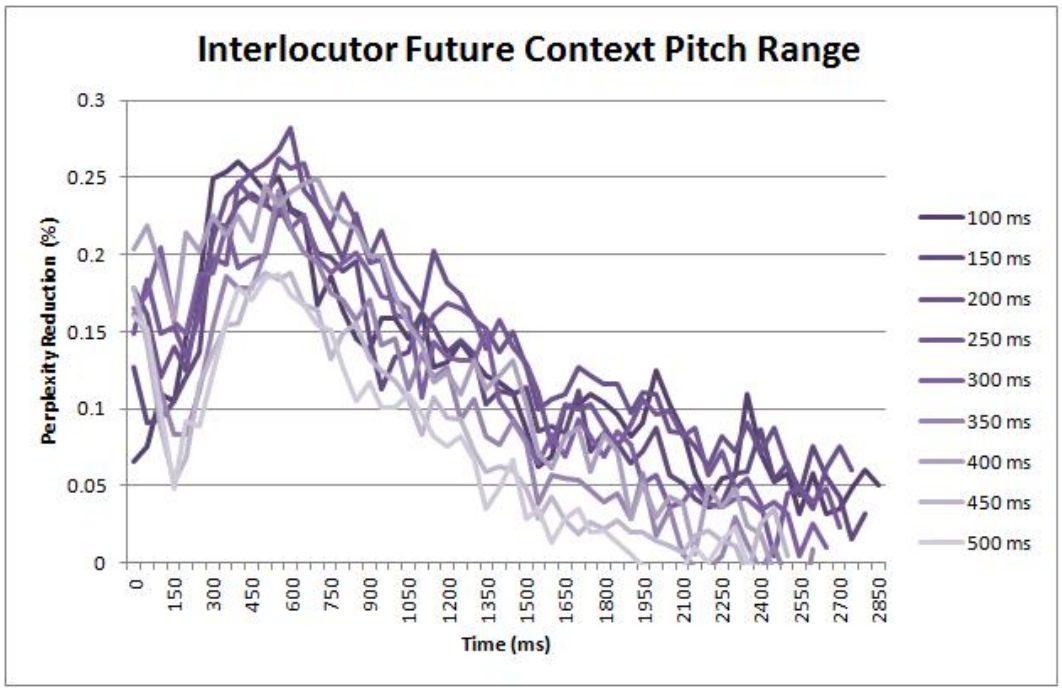

Figure 5.19: Perplexity reduction for interlocutor future pitch range models

Speaking rate feature benefits, shown in Figure 5.20, peak at the point-of-interest and rapidly fall below 0.1% after 150 ms for smaller width features. Benefits for longer width features (features over 250 ms) degrade faster, as they fall below 0.1% past 100 ms from the point-of-interest.

In general the trends in perplexity reduction produced by conditioning on past and future interlocutor features support the idea that interlocutor features are more beneficial for word prediction when they are referenced at times farther from the point-of-interest. This suggests that words may depend on interlocutor events that happen some time before or after the word occurs, as one would expect from typical human reaction times.

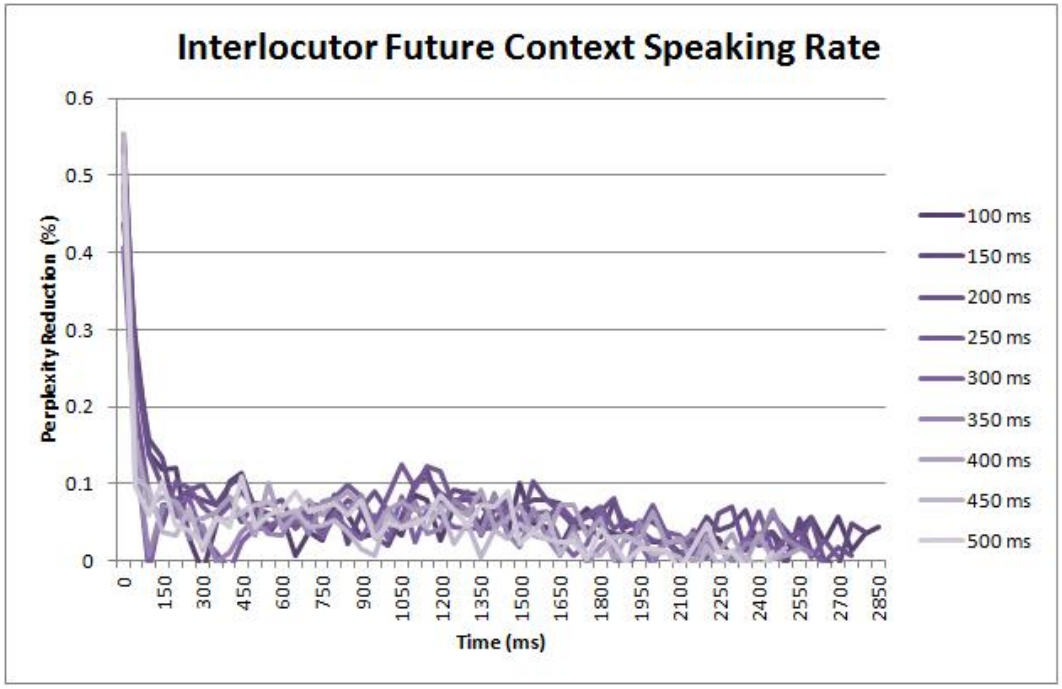

Figure 5.20: Perplexity reduction for interlocutor future speaking rate models

5.3 Feature Selection

Given the results from section 5.1 and 5.2, I can now make an informed decision on which features should be used for a comprehensive language model, as will be discussed in the next chapter. Taking the feature set of that model as a template for selection, I define a feature set that has:

- **Granularity** - Feature window sizes at certain context points are determined by the amount of information a certain feature contributes to the prediction task. Feature window sizes in a region of high contribution are generally finer than in regions where contribution is smaller.

- **Zero Gaps** - There is no blank space between features in the context space. This enables the model to capture all the information within the predefined context window.

- **Optimality** - When deciding between feature window sizes in a given context space, the window with better performance is selected.

- **Minimized Overlaps** - For some context spaces, there may be some overlap due to how features are selected. Although this may give the feature set some redundant information, the amount of overlap is kept to a minimum; in any case, redundancy will be filtered out by PCA when it is performed on the feature set.

Thus, for this feature set, features spanning regions outside of the high contribution regions in the context space are coarsely aggregated. The aggregation scheme is the same as in [12]. That is, after a fine-grained region of features calculated over a given window size, adjacent features are aggregated in multiples of two. For example, if the fine grain region is composed of 100 ms wide features, I aggregate and average the next two windows, producing the next adjacent feature. For the subsequent features, I apply the same operation over four windows, then eight so on and so forth until I cover the remainder of the context window space.

Pitch height and range features are treated differently. PCA requires that each feature has a real value. To avoid invalid pitch points, I simply replace invalid values with the average pitch range/height value over the corpus. I then apply normalization to the features and execute the aggregation scheme described above. There are also feature window size limits that need be imposed for pitch features. Respond appears to require at least 100 ms of context to reliably compute pitch height and range features. Therefore, the pitch features selected are only those calculated over windows of 100 ms or wider.

5.3.1 Speaker Features

As seen from the perplexity reduction results for speaker features, the majority of information lies at contexts close to the point-of-interest. For these features, I confine the fine-grained regions to range from the point-of-interest to a maximum fine context limit. I define this context limit for each feature and apply aggregation to features past this context

limit. For certain features though, I also define maximum context limits that are less than the predefined three second context window. This is because perplexity benefits seen further out are so small (below 0.1%) that they are deemed not important to word prediction. For these features, I impose a maximum context limit to avoid the introduction of noisy.

Volume

For past volume features, the fine region encompasses context from word onset up to 200 ms to the past. Due to the performance benefits seen from 50 ms windows in this context space, feature window sizes are taken at 50 ms within this space. For fine grain features, this results in four fine-grained windows close to the point-of-interest. For 200 ms and beyond, features are aggregated, resulting in five coarse-grained features. Future speaker volume's fine region lasts from 0-100 ms from word end time with features calculated over 50 ms, resulting in two fine windows. Coarse-grained windows after that point total up to five coarse features as well. The layout of these features in their respective context space is shown in figures 5.21 and 5.22.

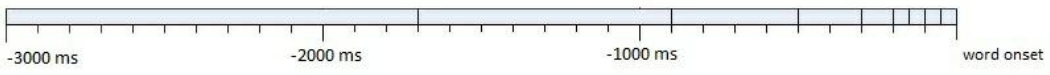

Figure 5.21: Speaker past volume features

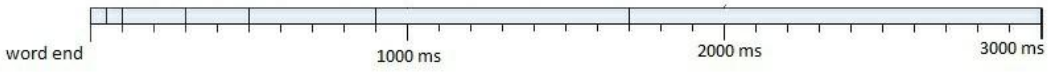

Figure 5.22: Speaker future volume features

Pitch Height

Past pitch height's fine region lasts up to 150 ms from word onset. Fine-grained features total up to two with features taken at 100 ms wide windows. Coarse-grained features after 150 ms from the point-of-interest total up to four features. Future height features have their fine region up to 250 ms producing three fine grain features (100 ms wide), and four coarse features after this region. The layout of these features in their respective context space is shown in figures 5.23 and 5.24.

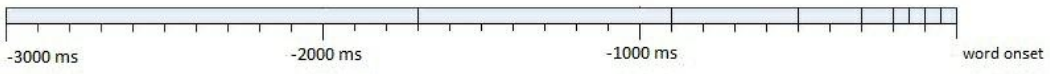

Figure 5.23: Speaker past pitch height features

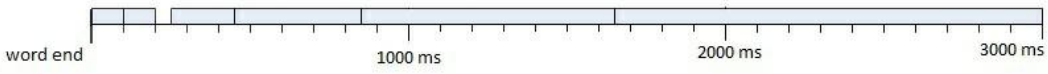

Figure 5.24: Speaker future pitch height features

Pitch Range

For past pitch height, the fine grain region happens between zero and 200 ms from word onset. Taking features at 150 ms wide regions, this produce two fine grain features within that space. To fill up the remainder of the context space, four coarse-grained features are produced. Future pitch range has its fine region from 0-250 ms, producing two fine grain features and four coarse feature in the remaining context space. The layout of these features in their respective context space is shown in figures 5.25 and 5.26.

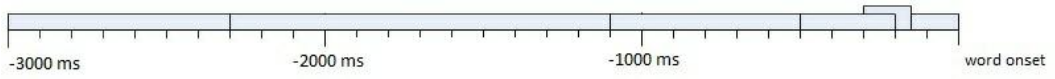

Figure 5.25: Speaker past pitch range features

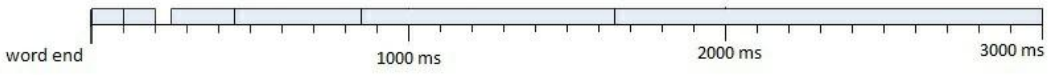

Figure 5.26: Speaker future pitch range features

Speaking Rate

Speaking rate's fine region lasts 200 ms from word onset, producing two fine grain features and four coarse grain features after that. The fine region for future speaking rate lasts from 0-100 ms from the point-of-interest. For this feature, I impose a 300 ms maximum context limit which produces one coarse grain feature 200 ms wide. The layout of these features in

their respective context space is shown in figures 5.27 and 5.28.

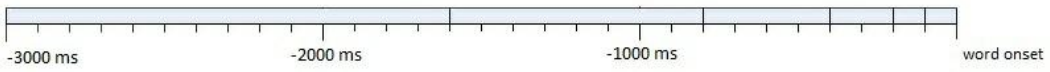

Figure 5.27: Speaker past speaking rate features

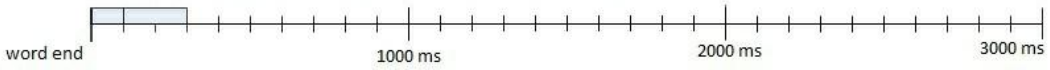

Figure 5.28: Speaker future speaking rate features

5.3.2 Interlocutor Features

For interlocutor features, the region of highest contribution appears some time before the point-of-interest for past features and after the point-of-interest for future features. Therefore, there exists a context space between the point-of-interest and the fine-grained region which I fill with a coarse-grained feature. This feature is selected based on best performance and minimum overlap with the fine feature region. Aggregation is still applied for features past the fine feature region. For certain features, I also impose a maximum context limit for features whose contributions are below 0.1% perplexity reduction.

Volume

For past volume features, the fine-grained region starts at 150 ms and lasts up to 550 ms, producing four fine grain features. The space between the point-of-interest and the fine feature region is filled with a 150 ms feature window, and the rest of the context space is filled by four coarse grain features. For future volume the fine region appears at 0-700 ms, producing seven fine features and four coarse features over the rest over the remaining context space. The layout of these features in their respective context space is shown in figures 5.29 and 5.30.

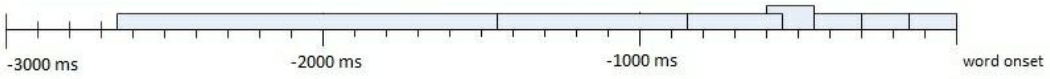

Figure 5.29: Interlocutor past volume features

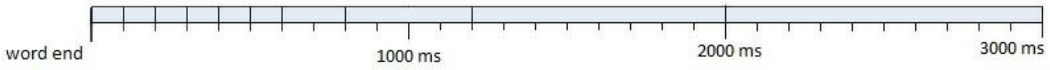

Figure 5.30: Interlocutor future volume features

Pitch Height

For past pitch height features, the fine feature region appears at 250 ms from word onset and lasts up to 600 ms, containing two 200 ms wide fine grain features. The space between the point-of-interest and the fine region is filled with a 250 ms wide coarse feature. Four

coarse features encompass the rest of the context. Future pitch height, whose fine feature region spans the space between 400 ms and 800 ms, produce four fine grain features. Coarse grain features total five, one 400 ms at the point-of-interest and four coarse features after the fine feature region. The layout of these features in their respective context space is shown in figures 5.31 and 5.32.

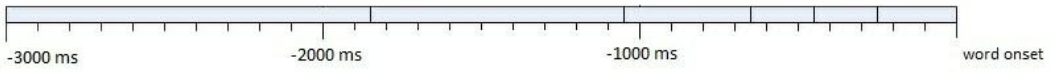

Figure 5.31: Interlocutor past pitch height features

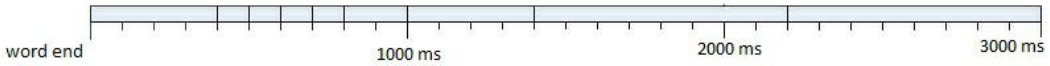

Figure 5.32: Interlocutor future pitch height features

Pitch Range

Past pitch range's fine feature region appears at the space starting at 150 ms from the point-of-interest and lasts up to 500 ms from the point-of-interest. Within this region, two fine features are used. For the rest of the space, a 150 ms wide feature is used at the point-of-interest and three coarse grain features after the fine feature region. For future pitch range, the fine feature region appears at 400-800 ms from the point-of-interest and encompasses two fine grain features. Coarse feature total four features: a 400 ms wide

window at the point-of-interest and three coarse features after the fine region. The layout of these features in their respective context space is shown in figures 5.33 and 5.34.

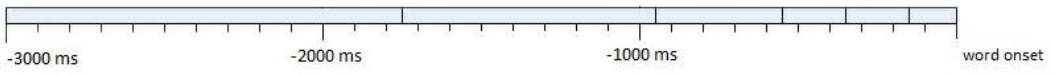

Figure 5.33: Interlocutor past pitch range features

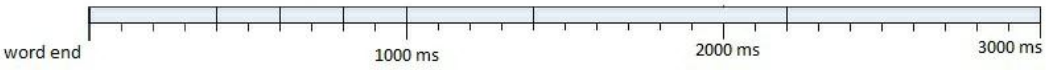

Figure 5.34: Interlocutor future pitch range features

Speaking Rate

Interlocutor speaking rate features exhibit different behavior from volume and pitch features. Their maximum benefit comes at the point-of-interest and then dwindles down after some time. For past speaking rate, the fine feature region lasts up to 400 ms and encompasses one fine feature. Two coarse grain features fill the remainder of the context space. For future speaking rate, I define the fine feature window lasting up to 100 ms from the point-of-interest and encompassing one fine feature. Four coarse grain features fill the rest of the context space. The layout of these features in their respective context space is shown in figures 5.35 and 5.36.

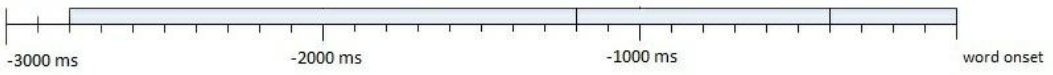

Figure 5.35: Interlocutor past speaking rate features

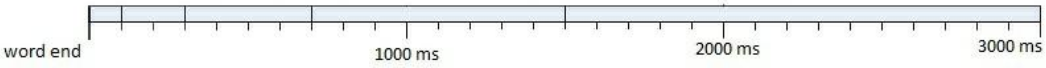

Figure 5.36: Interlocutor future speaking rate features

5.3.3 Selected Features

In total, for the six-second context space using both speaker and interlocutor feature, the feature set includes a total of 103 features. Broken down into each feature type, the set comprises 36 volume features, 28 pitch height features, 24 pitch range features, and 15 speaking rate features.

Chapter 6

Principal Component Language Model

In this chapter, I present the perplexity benefits gained by encoding the feature set defined in chapter 5 with Principal Component Analysis (PCA). One of the main gains of using PCA on these features is the production of an orthogonal set of features. The prosodic features included in the feature set, namely volume and pitch features, have a large amount of correlation between them. PCA does away with correlations between the resulting PCs, producing a feature set with no redundant information.

The resulting features are real valued features, thus binning is a necessity. Similar to the way that I binned feature values for the models in previous chapters, I bin PC values based on quartile thresholds, thus binning the values into four discrete categories: low, low-mid, mid-high, and high. Once binned, I can condition word probabilities based on their corresponding PC context and tweak word probabilities in the manner discussed in section 3.2.1.

I use Matlab's implementation of PCA on the prosodic features. Due to Matlab's local memory space limitations, I cannot perform PCA on all of the training set. I apply PCA to

20 different tracks of the Switchboard training set. Taking a sample every 10 milliseconds, this produces about 600,000 data points. At each of these points, I calculate the prosodic feature set defined in the previous chapter. PCA applied at each of these points produces a vector of 103 PCs at every 10 ms in a conversation.

6.1 PC Language Models

Table 6.1 presents the perplexity reduction for the top 15 principal component models when used in isolation. Contrary to the results from [12], where only three out of the first 10 components appear in the top 15 performing models, a good portion of the top performing components belong to the first 10 principal components. These first 10 components capture 51% of the variability seen in the data.

I set a 0.5% cut-off point on each PC model for use in the combined model. This results in the inclusion of 22 PC models. PC8 produced the largest amount of perplexity reduction at 2.42%, with PC91 close behind at 2.23%. The least performing model included is PC41 with 0.60% perplexity reduction.

Using these top 22 PC models I build a combined PC model that incorporates all of this information into a tri-gram language model. Table 6.2 shows the perplexity reduction for this combined model. The table also shows the effect that dataset size has on PC models that are trained on a smaller set of data. The reduced models models were a first iteration for PC modeling on the selected feature set. For these reduced models, I apply PCA to 450,000 data points. Although that is a significant amount of data, the

Table 6.1: Top 15 components and their respective perplexity reductions to the trigram baseline.

PC	Perplexity Reduction in Isolation	Weight in Tuned Combined Model
PC8	2.42%	0.8
PC91	2.23%	0.8
PC99	2.02%	0.7
PC7	1.46%	0.6
PC3	1.46%	0.7
PC10	1.43%	0.6
PC1	1.32%	0.4
PC5	1.26%	0.7
PC34	1.22%	0.4
PC9	1.12%	0.7
PC29	1.00%	0.7
PC49	0.99%	0.7
PC14	0.98%	0.6
PC4	0.91%	0.4
PC2	0.88%	0.5

models trained on a larger data set produce better perplexity results for individual PC models and, ultimately, the combined models. When used with default tuning parameters (k=0.3), the model achieves 19.42% perplexity reduction from its trigram baseline, not far from the 25.4% reduction one would expect by summing the individual component reductions. When using the same number of components as [12], the combined PC model produces reductions closer to the ones seen in that previous study

After tuning, I use the weights defined in Table 6.1 to produce a model that gives a 25.9% perplexity reduction from the trigram baseline. Unfortunately, the model falls short from the performance benefits produced by the PC model in [12] by a difference of –1.2% relative to the naive model. This is shown in Table 6.3. If the model is extended to 25 component models, the best that can be done is to match the performance of the naive PC model.

Table 6.2: Perplexity reduction for combined PC model.

Model	**Perplexity Reduction**
20 components, default weights, reduced	18.4%
22 components, default weights	19.4%
25 components, default weights, reduced	20.2%
25 components, default weights	21.2%

Unfortunately this model does not improve on the perplexity gained by the model in [12]. Although my model falls short from that of a previous study, it provides benefits for

Table 6.3: Perplexity reduction for tuned combined PC models.

Model	**Perplexity Reduction**
PC Model (Ward and Vega, 2012)	26.8%
Top 22 components	25.9%
Top 25 components	26.8%

76% of the words in the test data. This, compared to the previous model's benefit on 66% of the words on the same data set, points to the possibility that this model's benefits are more reliable but not often extreme enough to make a big difference for prediction.

Chapter 7

Discussion

In this chapter, I discuss and analyze features of the model evaluated in the previous chapter. I also take a look at the possible meanings behind each of the principal components obtained, to see how they are relevant to language modeling.

7.1 Model Benefit Analysis

In this section, I analyze instances in the evaluation where the combined PC model affected baseline probabilities positively and negatively. This is done in the same manner as in [14]. I listened to 10-second-long audio clips centered around these instances and payed close attention to speaker and interlocutor actions in these dialogue excerpts.

The combined PC model provides benefit to 76% of the words seen in the Switchboard test set. Among these instances are spots where the speaker is talking fluently and with little to no interaction from the interlocutor. The PC models are robust at instances of overlap between speaker and interlocutor. The PC models even do well at places where the speaker is disfluent. These instances are mainly self-repeats and self-fixes, places where a language model usually fails. The combined PC model captures enough information to boost the

correct word probabilities at these places. Among the instances that were benefitted are filler words, backchannels and content words like *privacy*, *couple* and *football*.

On the remaining 24% of the cases there are some interesting patterns. The combined PC model is sensitive to the use of the mock-quoting *oh*. The quoting *oh* here is defined as the use of the word *oh* to start a story from previous experience. An example of this is illustrated in Table 7.1. In the first sample utterance, the PC models penalize the use of this word in that particular context. Only five PC models provide S-Ratios that are 1.00 or greater. The rest of the PC models give lower S-Ratio estimates, sometimes as low as 0.59.

Table 7.1: Examples of mock-quoting *oh*.

Utterance samples
. . . and I'll say *oh* about eight kilometers . . .
. . . and you know *oh* golly it's a metric . . .
. . . find out you know *oh* this kid may have a . . .

In terms of lexical context, the mock-quoting *oh* is problematic because it is a word that can appear at any point in conversation where a previous experience is told. The lexical context before these instances is varied and is often not discriminative enough to pinpoint the places the quoting *oh* will appear. The same can be said for the prosodic context around these instances. The prosody around the word may not be discriminative enough, leading the PC models to give low probability ratios to this word, ultimately hurting the

baseline model at these points.

Another place where the model penalizes words is where content words are used in one-word utterances or as the first word in an utterance. One-word utterances like *management* or utterances starting with *lawn* or *great* were among the instances where the model did not do well.

Table 7.2: Example utterance of model failure.

Speaker	**Utterance**
Interlocutor	. . . at being active he will go into uh
Speaker	**Management**
Interlocutor	managing and yeah you know . . .

7.2 Dimension Analysis

A feature of PCA is that it is able to capture information relevant from the features to which it is applied. Although each component may have captured information about speech, the nature of that information is unknown. To understand the dialogue state characteristics that influence the occurrence of particular words in spoken dialogue requires further analysis.

To try and identify this information, I listened to points in the corpus that are described by high and low points in each of the dimensions relevant to language modeling.

This method is the same as the one in [11]. My interpretations of these dimensions are completely inferred from the data, however there is a good amount of subjectivity in these interpretations.

The process for inferring these definitions is the following. I listened to the six seconds of audio surrounding the point of interest, taking into account the actions of the speaker, the actions of the interlocutor and the prosody produced by both speakers. The interpretations discussed in this section were done on a set of dimensions generated on a reduced data set for PCA, hence the numbering of the dimensions is unrelated to the ones mentioned in the previous chapter.

Dimension 89 - Interlocutor Engagement

Low points in this dimension characterize speech that is dominated by the speaker of interest. An example is shown in Table 7.3. These points are characterized by instances of speech where the speaker is in a monologue, either making a point or telling a story. The interlocutor has little to no involvement at these points in the conversation. Their only role at these points is strictly that of a listener whose only contributions are back-channels. At high points, while the speaker of interest is still owning the turn, these areas of speech follow interlocutor actions (Table 7.4). The interlocutor often makes a question or statement to which the speaker responds accordingly. Thus, I identify this dimension as one that characterizes interlocutor engagement in the conversation.

Table 7.3: Example for dimension 89: Low dimension values.

Speaker	**Utterance**
Speaker	. . . bad team and they were mean we had so many penalties and **it** gets really you just
Interlocutor	(back-channeling)

Table 7.4: Example for dimension 89: High dimension values.

Speaker	**Utterance**
Interlocutor	I do I work for the school system
Speaker	ah
Interlocutor	Richardson school
Speaker	well that's a **pretty** large corporation
Interlocutor	yes it is

Dimension 8 - Speaker Floor Grab

Low points in this dimension characterize speech were the speaker of interest makes responses to interlocutor points and statements. These responses are short in nature and go straight to the point. The speaker's intention at these points is to produce the response and yield the floor back to the interlocutor so that he/she can keep on talking. High points in this dimension though, show the opposite. At these instances, the speaker grabs the turn and holds it for a longer amount of time. Some of these points show the speaker taking the turn at a point where he did not expect to be talking. Another instance shows the speaker reinitiating a previous topic, as the topic at hand expires. Thus, this dimension is identified with the speaker's intention to keep on talking. Examples are shown in tables 7.5 and 7.6.

Table 7.5: Example for dimension 89: Low dimension values.

Speaker	**Utterance**
Interlocutor	. . . things either
Speaker	oh [laughter]
Interlocutor	so
Speaker	what uh what kind **of**
Interlocutor	well I just finished . . .

Table 7.6: Example for dimension 89: High dimension values.

Speaker	**Utterance**
Interlocutor	I'll tell you another good book do you like scary things?
Speaker	**um** well ...

Dimension 99 - Information Addendum

Low points in this dimension are characterized by areas of speech where the speaker of interest adds information relevant to the topic at hand. The nature of this information is such that it advances the conversation further. These speech excerpts include points where the speaker states information that contradicts a point made earlier by the interlocutor. High points are also characterized by the addition of the information by the speaker; However, the nature of this extra information is different. This information is usually just extra information, mainly for the sake of drawing out the speaker's own turn or keeping the floor for a bit longer. Thus, I identify this dimension as one where the speaker contributes redundant vs. new information. Examples are shown in tables 7.7 and 7.8.

Table 7.7: Example for dimension 99: Low dimension values.

Speaker	**Utterance**
Interlocutor	... you know something in
Speaker	you know the plano newspaper each each **day** in fact has a a little list

Table 7.8: Example for dimension 99: High dimension values.

Speaker	**Utterance**
Speaker	somewhere and look at it but uh i don't seems like **it** takes too much time or something
Interlocutor	well that's true

Dimension 10 - Showing Interest

Low points in this dimension characterize areas of speech where the speaker of interest seems withdrawn from the conversation. These instances include parts of the conversation where the interlocutor shows concern for the safety of family members, to which the speaker responds with a flat and quiet "well good", shown in Table 7.9. Another point in the conversation shows the speaker going back to a previous topic after the interlocutor tries to make a joke. The high points describe parts of the conversation where the speaker makes a joking statement out of their own experience (Table 7.10), in an attempt to increase the atmosphere of the conversation. Thus, I identify this dimension as showing interest on the conversation.

Dimension 3 - Speaker Engagement

Low points in this dimension are described by points in conversation where speaker sounds disinterested and inactive in the conversation. High points describe areas of conversation

Table 7.9: Example for dimension 10: Low dimension values.

Speaker	**Utterance**
Interlocutor	. . . they're either on their way back or just got back from Georgia so
Speaker	**well** good
Interlocutor	hopefully they missed most of the rain

Table 7.10: Example for dimension 10: High dimension values.

Speaker	**Utterance**
Speaker	we grew up on a farm so we just had to play with each other cause **there** was no one else out there [laughter]
Interlocutor	[laughter] oh yeah

where the speaker is confident about his words and is clearly engaged in the conversation with the interlocutor. Examples are shown in tables 7.11 and 7.12.

Table 7.11: Example for dimension 3: Low dimension values.

Speaker	**Utterance**
Interlocutor	. . . huh
Speaker	but they did something a little different with it **yeah**
Interlocutor	uh-huh well that I would be interested in

Table 7.12: Example for dimension 3: High dimension values.

Speaker	**Utterance**
Interlocutor	well I guess they would I don't know why they wouldn't
Speaker	well I've never seen **one**
Interlocutor	oh you haven't?
Speaker	I've never seen one
Interlocutor	I just thought it was my set that . . .

Chapter 8

Conclusions and Future Work

The work done in this thesis shows that careful selection of prosodic features had no positive effect on the performance over a model that incorporates this information naively. Although no improvements were made in terms of performance, I have discovered which features contribute the most information to the word prediction task for language modeling. The redeeming quality of this model is that it benefits a larger fraction of the words in the evaluation corpus than the naive PC model. This fact says that basing feature selection on this information may be a step in the right direction. Thus, it is worthwhile to refine the approach in this thesis.

The only problem left to solve is to figure out how to capture this information effectively in a feature set. The naive model provides benefits of greater magnitude than the model in this study does. This could be due to the inclusion of noisy data in the coarse-grained regions, due places where the selection method was not refined. Once refinement is done, this information can then be applied to the selection of prosodic features for future research efforts. Although my work for this thesis ends here, there is ample room for improvement and extension of techniques for the application of prosodic features to language modeling.

8.1 Improvements

For this thesis, I used percent perplexity reduction as the sole factor for assessing the information contribution for a given prosodic feature. While it is good indicator, other factors can be used in the process for defining that a prosodic feature is good. One of these factors is word benefit percentage. I can use the ratio of words benefitted to total words to determine if a given feature hurts the language model more than it benefits word probabilities.

Feature aggregation is done for features calculated past fine-grained regions so that information included in contexts further from point of interest is included in the model. The inclusion of such features is done in hope that they would add some normalizing information. However, their utility is not known for sure. Thus, an anlysis of the worth of these features should be done to determine their contribution to the word prediction task to improve the model proposed in this thesis.

8.2 Future Work

In this study, I set a maximum size for the MCW to be 500 ms. However, it seems worthwhile to evaluate MCWs that are longer than 500 ms. The results on conditioning word probabilities over longer MCW features will provide insight on the effect of longer context windows at further points in the context space.

As mentioned previously, PCA has the ability to encode information not directly seen

by each of the prosodic features individually. These components may capture information that may be beneficial for their use not only on the word prediction task, but also for other aspects of dialog (e.g., end pointing, word spotting, hot-spot discovery, etc.). Analysis on the remaining top PCs to find their meaning in dialog space is left for future work.

Although perplexity is a good indicator of the performance of a language model, real performance gains produced by a modified model are better measured when the model is used in a speech recognizer. Thus, the application of this model in a speech recognizer should determine the model's real contribution to an ASR system.

Although the main goal in this work was not met, a big first step into effectively capturing prosodic information for language modeling has been taken. With this work as a stepping stone, future research can improve on this work and find the best way of incorporating this information to improve language models, automatic speech recognition, and, ultimately, systems that make use of speech technology.

References

[1] Sankaranarayanan Ananthakrishnan and Shrikanth Narayanan. Improved speech recognition using acoustic and lexical correlates of pitch accent in a n-best rescoring framework. In *ICASSP*, pages 873–876, 2007.

[2] Ken Chen, Mark Hasegawa-Johnson, and Jennifer Cole. A factored language model for prosody-dependent speech recognition. In Michael Grimm and Kristian Kroschel, editors, *Robust Speech Recognition and Understanding*, pages 319–332. I-Tech, 2007.

[3] J. J. Godfrey, E. C. Holliman, and J. McDaniel. Switchboard: Telephone speech corpus for research and development. In *Proceedings of ICASSP*, pages 517–520, 1992.

[4] J. Hamaker, Y. Zeng, and J. Picone. Rules and guidelines for transcription and segmentation of the Switchboard large vocabulary conversational speech recognition corpus, version 7.1. Technical report, Institute for Signal and Information Processing, Mississippi State University, 1998.

[5] Songfang Huang and Steve Renals. Modeling prosodic features in language models for meetings. In A. Popescu-Belis, S. Renals, and H. Bourlard, editors, *Machine Learning for Multimodal Interaction IV (LNCS 4892)*, pages 191–202. Springer, 2007.

[6] Elizabeth Shriberg and Andreas Stolcke. Prosody modeling for automatic speech recognition and understanding. In *Mathematical Foundations of Speech and Language*

Processing, IMA Volumes in Mathematics and Its Applications, Vol. 138, pages 105–114. Springer-Verlag, 2004.

[7] Andreas Stolcke. SRILM – an extensible language modeling toolkit. In *Proc. Intl. Conf. on Spoken Language Processing*, 2002.

[8] Klára Vicsi and György Szaszák. Using prosody to improve automatic speech recognition. *Speech Communication*, 52:413–426, 2010.

[9] Nigel G. Ward and Alejandro Vega. Modeling the effects on time-into-utterance on word probabilities. In *Interspeech*, pages 1606–1609, 2008.

[10] Nigel G. Ward and Alejandro Vega. Towards the use of inferred cognitive states in language modeling. In *11th IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 323–326, 2009.

[11] Nigel G. Ward and Alejandro Vega. A bottom-up exploration of the dimensions of dialog state in spoken interaction. In *13th Annual SIGdial Meeting on Discourse and Dialogue*, 2012.

[12] Nigel G. Ward and Alejandro Vega. Towards empirical dialog-state modeling and its use in language modeling. In *Interspeech*, 2012.

[13] Nigel G. Ward, Alejandro Vega, and Timo Baumann. Prosodic and temporal features for language modeling for dialog. *Speech Communication*, 54:161–174, 2011.

[14] Nigel G. Ward, Alejandro Vega, and David G. Novick. Lexico-prosodic anomalies in dialog. In *Speech Prosody*, 2010.

Curriculum Vitae

Alejandro Vega was born in C.D. Juarez, Chihuahua, Mexico on December 14, 1988. Born as the second of two children from Candelario Vega and Rosa Maria Ramirez, Alejandro pursued a Bachelor's degree at the University of Texas at El Paso in the year 2006. During this time he spent three years working under the supervision of Dr. Nigel G. Ward as an undergraduate research assistant for the Interactive Systems Group. With his research on language modeling, Alejandro co-authored 4 major publications along with a journal article. His work helped in securing a $500,000 research grant from the National Science Foundation. In 2010, Alejandro received his bachelor's of science degree and pursued his Masters under the supervision fo Nigel Ward. In December 2012, Alejandro became the first in his family to successfully obtain a Masters degree.