USING EMOTION AS INFERRED FROM PROSODY IN LANGUAGE MODELING

SHREYAS ASHOK KARKHEDKAR

Department of Computer Science

APPROVED:

Nigel Ward, Chair, Ph.D.

David Novick, Ph.D.

Stephen Crites, Ph.D.

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

to all of

my family

with love

USING EMOTION AS INFERRED FROM PROSODY IN LANGUAGE MODELING

by

SHREYAS ASHOK KARKHEDKAR, M.Tech.

DISSERTATION

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

DOCTOR OF PHILOSOPHY

Department of Computer Science

THE UNIVERSITY OF TEXAS AT EL PASO

May 2013

Acknowledgments

I would like to thank The Almighty for making this day possible. I would like to thank my family, here in US and back home in India, for the love and support they have provided throughout my life.

I would like to give my deepest thanks to Dr. Nigel Ward for his help as an advisor and support as a mentor. I thank Dr. David Novick for his teachings and feedback over these past four years. I thank Dr. Stephen Crites for his comments and being on my committee.

I would like to thank all my friends for all the good times I have had. I thank all the members of ISG – present and past. I would also like to express my gratitude towards the people working at Research Cloud @ UTEP.

This dissertation would be incomplete without mentioning Alejandro Vega. His work on language model evaluation laid a concrete foundation for me. Over these years, he has helped me, both personally and professionally, and I am deeply indebted to him.

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

Abstract

Research has focused on using prosody as an alternative source of information for language modeling. However, prosody is a surface phenomenon and to develop deeper models of language production, the underlying mental processes need to be considered. There are several cognitive factors, such as dialog-states and formulation, that have been given attention. However, emotion – as a cognitive factor, has been neglected so far.

Speakers' emotional state plays an important role in spoken dialog. Participants seem to infer each others emotional state from multiple cues and react accordingly. In particular, these states manifest themselves moment-by-moment in the speakers voice. This dissertation attempts to model these changes and use them to improve word probability estimates for language modeling.

A small set of conversations from the Switchboard corpus was labeled for emotion. Rather than using a class-based approach, I have used a dimension-based approach, to account for the subtle changes in emotion that occur in spontaneous dialog. I developed several models using different machine learning techniques that estimate the emotion value on each dimension independently. Once the speaker's emotional state is recognized, the probability estimates of words that the speaker might say next are refined based on that recognized state. Using emotion for language modeling in this way resulted in a 1.35% reduction in perplexity over the baseline.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Recently, the use of automated systems in everyday life has been increasing. In particular, interactive spoken dialog systems have become commonplace with improvements in speech recognition technology.

Automatic Speech Recognition (ASR) technologies mainly process speech at the signal level. However, research is focusing more on developing models of speech production and recognition that involve processing at levels closer to the cognitive level of human speech production and recognition [25][31]. Although, the cognitive underpinnings of speech production and recognition remain hidden, they manifest themselves through the speaker's expressions – words, gestures and intonation. In particular, speaker's emotional state is directly influenced by the underlying cognitive processes [23]. Figure 1.1 shows a schematic diagram of how cognitive processes are involved in speech production. Cognitive processes such as syntactic processing, semantic processing and emotion affect cognitive-level states such as dialog control and emotion. The changes in these states manifest themselves as signal-level surface events such as changes in prosody and word choice.

Knowledge of the speaker's current emotional state and how to respond to it has a positive effect on the quality of interaction [1]. However, there are other possible ways to use the knowledge of the speaker's current emotional state — in particular, in predicting the speaker's upcoming words.

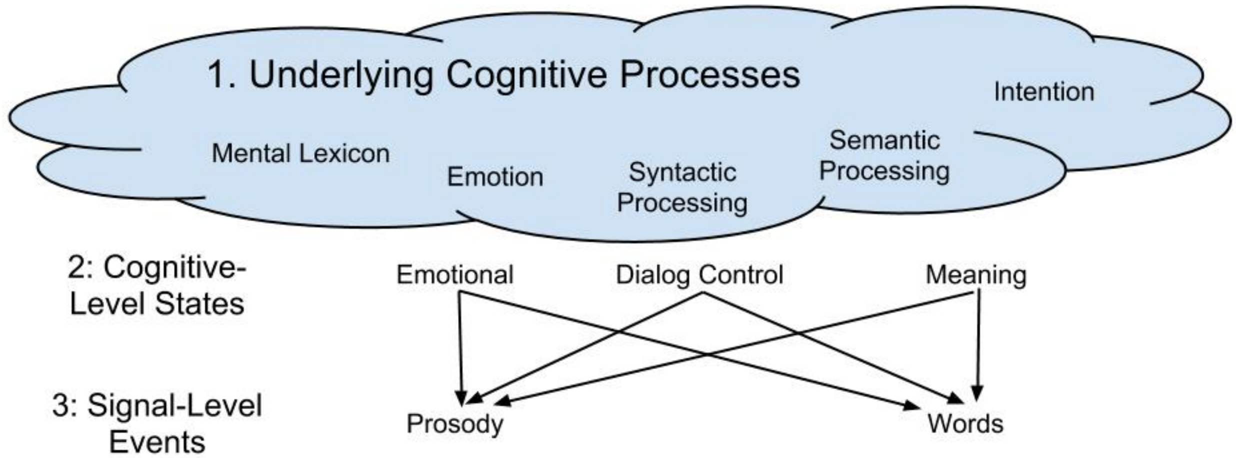

Figure 1.1: Schematic representation of some relationships between cognitive processes and speech production.

1.1 Aim

In this dissertation I focus on using speaker's current emotional state as means of improving speech recognizers. In particular, I focus on improving language models to better predict what the speaker might say next by modeling word choice as a function of the speaker's current emotional state.

1.1.1 Language Modeling

Figure 1.2 shows a schematic diagram of a speech recognizer. The speech recognizer comprises two modules: an acoustic model (AM) and a language model (LM). The acoustic model comprises the statistical representations for phonemes using feature vector sequences. The LM generates probability estimates for word sequences given a context. These probability estimates are used by the decoder to choose from the set of possible recognitions and select the most probable candidate.

The performance of a speech recognizer can be improved by improving either of the two models. This research focuses on improving the LM by using information on the speaker's

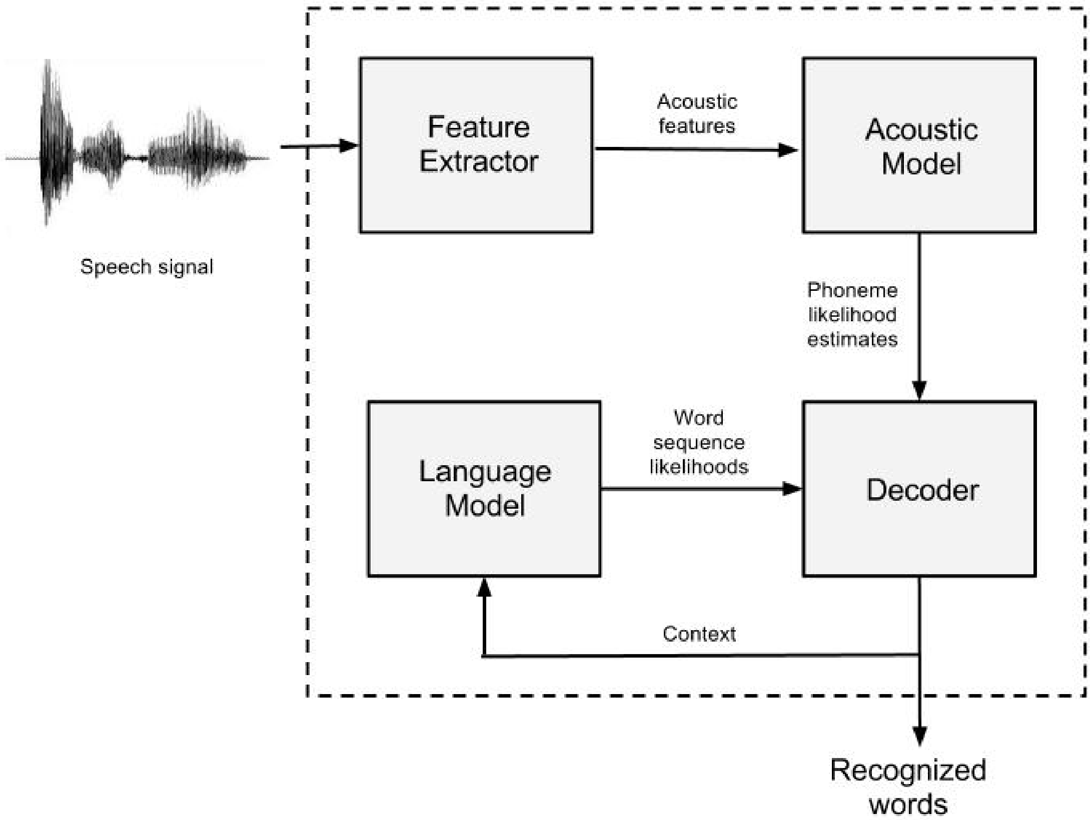

Figure 1.2: Schematic representing the modules of a speech recognizer.

emotional state.

1.1.2 Emotional State

Emotion plays an important role in human-human communication. People seem to infer and react to each others emotion in their speech using spoken content and prosody. It seems obvious that the speaker's emotion plays some role in deciding what he or she might say next. For example, if the speaker is happy, then he or she might start laughing or utter words while laughing. However, if the speaker is sad, then he or she might use words such as "school." Emotion states tend to correlate with usage. Emotion words correlate positively with usage of pronouns, auxiliary verbs and negations and emotion words correlate negatively with usage of articles, prepositions and relativity words [36].

In this research, I attempt to find relationships between words and emotional states and to develop models that capture these relationships and use them for language modeling.

1.2 Thesis Statement

My main hypothesis is that augmenting a language model with information related to the speaker's current emotional state will improve performance. To test this hypothesis, I build a language model that adapts to the speaker's current emotional state by adjusting the estimates of emotionally-appropriate words; I compare the model with a baseline model that does not use the speaker's emotional state.

My second hypothesis is that this emotion-adaptive language model will perform better than a language model that uses raw prosodic information. To test this hypothesis, I compare the performance of the emotion-adaptive language model, built to test the first hypothesis, with the prosody-based language model described in [49].

The rest of the dissertation is structured as follows:

Chapter 2 presents the literature related to language modeling, the use of para-linguistic features in language modeling, recognizing emotion and the use of emotional states in language modeling.

Chapter 3 describes the process of developing models for emotion recognition and the evaluation of their quality.

Chapter 4 describes the process of using emotion recognizers in a language model and presents the results obtained by using an emotion recognizer in a language model.

Chapter 5 discusses the results obtained from using an emotion-augmented language model and presents the analysis of words which typically occur in various emotional contexts.

Chapter 6 presents the conclusions from this research and charts out directions for the future in this field.

Chapter 2

Related Work

This chapter reviews literature related to language modeling, use of prosody in language modeling, detecting emotion from prosodic features, and the use of emotion in the field of language modeling.

2.1 Language Modeling

A language model generates probability estimates for the next word, which occurs in the vocabulary V of the language model, given a context. Traditionally, language models have used statistical information from textual training data where the probability of the next word is conditioned on the previous words (Equation 2.1).

$$P(w_i) = P(w_i|w_{i-1}, w_{i-2}, ...); \forall k, w_k \in V \tag{2.1}$$

Researchers have used several methods to capture statistical information from textual data. For example, Jelinek *et al.* [18] limited the context in equation 2.1 to n - *1* previous words. Such models are commonly known as n-gram models. Bahl *et al.* [3] used a decision tree-based approach to generate the probabilities for the next word. Pietra *et al.* [10] developed an adaptive technique using minimum discriminant estimation.

Although, there are several techniques to build language models using statistical information from text alone, the state-of-the-art language models are based on n-grams [28]. However, models that use lexical information alone have two major limitations. First, they require a large amount of training data to generate reliable estimates. And second, the benefit of having large data sets seems to stagnate after a certain point [34]. Looking at alternate sources of information for improving LMs, therefore, becomes imperative.

2.2 Using Prosody for Language Modeling

Non-verbal speech features provide an alternative source of information for improving language models. In particular, considerable effort has been put into extracting and using speech prosody for language modeling. Figure 2.1 shows a schematic diagram of a speech recognizer where the language model uses prosodic information, rather than relying solely on the lexical context as shown in Figure 1.2.

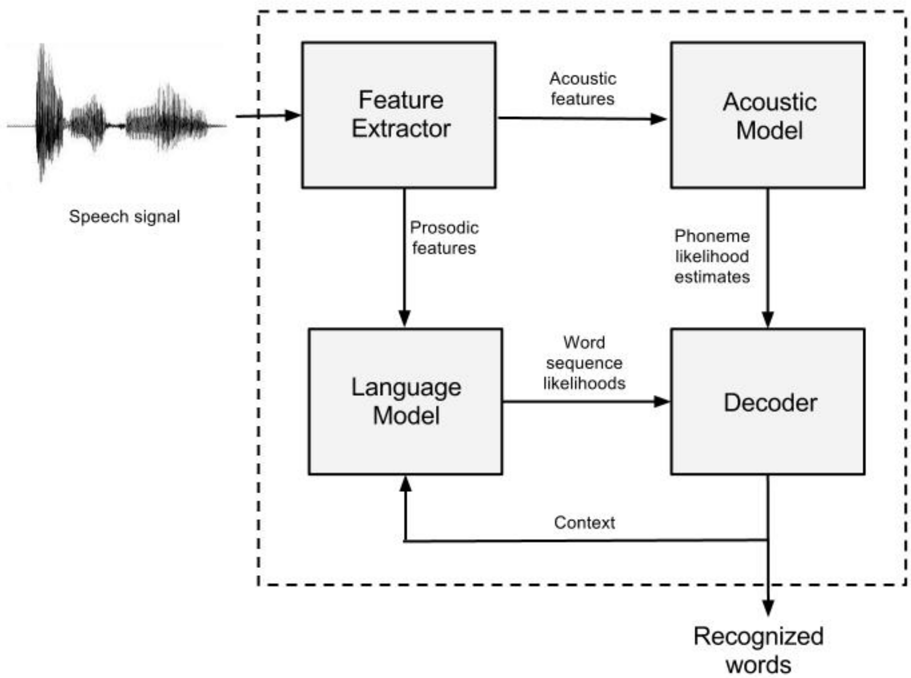

Figure 2.1: Schematic diagram representing a language model using prosodic features to generate word sequence likelihoods.

Stolcke *et al.* [42] treated prosody as information revealing "hidden-events" in the speech stream. These hidden-events could be speech disfluencies such as filled-pauses, repetition and deletion etc. They modeled speech as a stream of tokens, where each token can be a word or a disfluency-event, and achieved an absolute reduction of 0.9% in word error rate.

Taylor *et al.* [44] used prosody to reduce word recognition error in spontaneous goal-oriented speech. They first classified each utterance into one of twelve classes of speech based on the prosody of the utterance. The classes are based on conversational dialog acts such as asking a question, acknowledging, making a request etc. Separate bi-gram models were then developed for each class. When used in a recognizer, the models gave an improved word recognition accuracy by 1.4% over a general language model. Using separate class-based n-grams in spontaneous speech present two problems. First, the number of classes have to be chosen *a priori*. However, choice of the number of classes is subjective. Second, class-based n-grams increase data sparsity. For example, a word's (w) occurrences might fall completely in one class.

To avoid these problems, Ward *et al.* [49] predict the probability of the upcoming word based on how common a word is in a given prosodic context. For example, they computed the probability of the word *I* in a high-volume, low-pitch height and slow-speaking rate region. They reported a 8.4% improvement in the performance of a language model in terms of perplexity by using simple combinations of prosodic features to augment the trigram model. Similarly, Karkhedkar *et al.* [20] used Gaussian mixture models to represent the typical prosodic contexts of different words. When used in a language model, they achieved a 2.28% reduction in perplexity.

Using prosody, as an additional source of information, has been beneficial for language modeling. However, extending the use of prosody for language modeling becomes computationally difficult as more context is used. For example, Vega [45] mentioned a feature space of more than 7000 features using only 4 prosodic features derived over different parts of a 6-second context. Thus, it becomes imperative to develop methods that effectively use information from surface prosody to model deeper cognitive processes behind speech production. Interpreting the speaker's emotional state from prosody presents one way of approaching the development of such models.

2.3 Emotion

Research on emotion in speech has primarily focused on its detection [22] and categorization. Recognizing emotion is important for human-system interaction as it can potentially make the interaction seem more natural. Real-world scenarios include spoken tutor systems [48], voice portals [7] and automated counseling-agents [1].

2.3.1 Describing Emotional State

Most research on emotion detection focuses on categorizing it into one of several classes. For example, Ekman [12] mentioned six basic emotion classes for English. These emotions are proposed as universal basic emotions [33]. More emotion classes arise from different "blends" of basic classes. Plutchik [32] presented a 3-dimensional emotional index cone for blending of basic emotion classes.

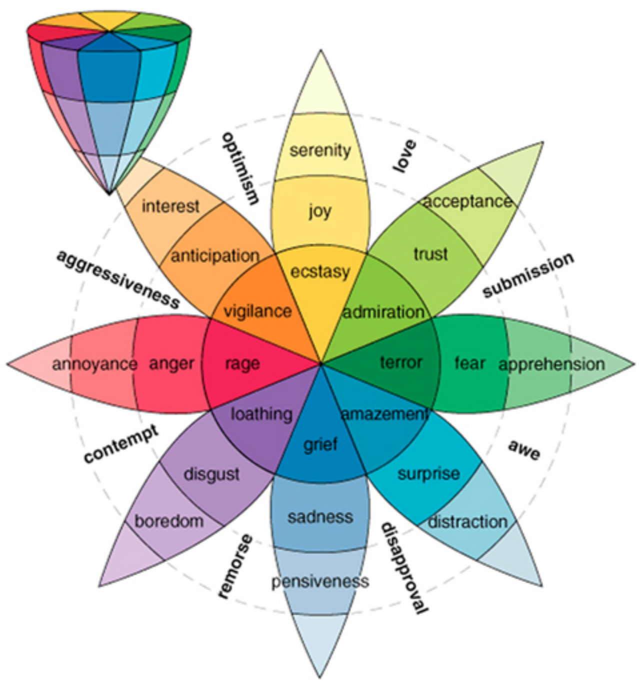

Figure 2.2: 3-dimensional emotional index as presented in [32].

However, in everyday interactions, people exhibit nonbasic, subtle, and rather complex mental and affective states such as thinking, embarrassment, or depression [4] [6]. Therefore, a single emotion class might show a larger variation in these complex states and might fail to properly exploit a rich source of affective information [35]. An alternate approach to categorizing emotions is to represent emotion in a space formed by a small set of continuous dimensions [52]. For example, Mehrabian *et al.* [26] proposed representing emotion as a 3-tuple of real values, comprising valence (positive/negative), arousal (active/passive) and power (dominant/submissive).

Another parameter of emotional-state description is the duration over which emotional state is determined. Research methods on emotion recognition often determine the emotional state over the duration of the utterance [1] [39] [13]. However, for this research, I need moment-by-moment annotation of emotion, as explained in section 3.1.

Nicolaou *et al.* [30], noting the shift towards subtle, time-continuous and context-specific recognition of emotion in real-world settings, used multimodal analysis to recognize emotion, in terms of valence and arousal, on a continuous scale. In particular, they extracted facial expressions, acoustic cues and shoulder gestures from the SAL database for predicting continuous value emotion dimensions independently. These independent predictions were then fed into a fusion network which computes the final prediction for each dimension. Using support vector regression (SVR), they reported a correlation of 0.419 for activation and 0.146 for valence using acoustic cues alone. These correlations are lower than those reported in other research [1]. This could be attributed to the fact that these continuous predictions are based on shorter context rather than the entire utterance. Despite their interest in subtle real-world emotions, they used a database of induced emotion rather than spontaneous and natural emotions.

In summary, there are several ways of representing emotion. Classical approaches have focused on small and subjective sets of emotion. Modern approaches have focused on representing emotion as a set of orthogonal and continuous dimensions. This facilitates blending of the traditional emotion classes and makes the representation of emotion more flexible.

2.3.2 Using Speaker's Emotional State in Speech Recognizers

Decoding the speaker's emotion represents one problem. Another challenge is to develop models of system behavior that adapt appropriately towards the the detected emotion. While various components could use emotion information, so far attention has been devoted primarily towards (1) the dialog manager module of a spoken dialog system to decide on an emotion-appropriate response and (2) the synthesis of emotion-appropriate speech. However, speech recognition systems that use emotion information are rare – even though there is evidence that emotional speech degrades the performance of a speech recognizer [47].

Researchers have investigated the negative impact of emotion on the accuracy of a recognizer and have suggested different methods to overcome the problem. For example, Steidl *et al.* [41] compared the performance of speech recognizers on motherese, emphatic and angry children's speech against their performance on neutral speech. In particular, they conducted two experiments. First, they compared the performance of a recognizer trained on neutral speech to recognizers trained specifically for the each of the three emotion classes. Second, they compared performance of a speech recognizer trained on emotionally-colored speech against a speech recognizer trained on neutral speech. To add emotional speech data for training, multiple copies of data from each of the three classes were augmented to the neutral data. Acoustic and language models were then trained on this emotionally-colored data and used in a recognizer. Their results showed that introducing only a small quantity of emotionally colored speech in training data improves the recognition of "angry" and "emphatic" speech. They concluded that recognition can be improved by adapting acoustic and linguistic models to emotional speech.

Schuller *et al.* [37] presented affect–adaptation techniques for acoustic models to improve the performance of speech recognizers on angry speech, using neural networks (NNs) and hidden Markov models (HMMs). By dynamically adapting the acoustic models to the speaker's emotion at the sentence level, they achieved a 16.59% reduction in word recognition error.

Vlasenko *et.al* [47] developed acoustic models on acted affective speech. When used for spontaneous emotional speech recognition these models gave a 25.43% improvement for word-accuracy over models that were trained on neutral speech.

Thus, speech recognizers that use an emotion recognizer can perform better than those without one. Studies involving using emotion recognizers for speech recognition have mainly focused on using it in acoustic modeling. Their use in language modeling is, however, largely neglected.

2.4 Using Emotion for Language Modeling

The only research effort to incorporate emotional behavior to improve language models focused on increasing the representation of tokens that convey emotion. In particular, this method trained more on words that occur in an affect lexicon. Athanaselis *et al.* [2] used the Whissel emotional dictionary [51] to augment a neutral corpus with affect-oriented sentences. They used the BNC corpus for baseline training. From this training set, sentences that include words belonging to the Whissel lexicon were extracted. These sentences were then appended to the baseline training set – thus, increasing the population of emotionally-colored sentences. A language model trained on this augmented set achieved an average improvement of 20.18% in recognition.

Although such improvement is significant, there are three main issues with this model. First, their training method was biased towards emotionally-colored word tokens. Their emotionally-augmented model trained on affect words 20% more and thus skewed the raw estimates. While increasing the raw counts of affect words improves their prediction, this will hurt the estimates of non-affect words. Therefore, the model becomes considerably biased towards affect words. Second, their test data seems to be inadequate in size, as they used the recognizer for roughly 600 tokens only. It would be interesting to know whether the emotion-augmented model sustains its level of improvement over larger test sets. Third, they completely ignored emotion from the input speech signal. Rather, they inferred emotion from the presence or absence of a word in a emotional dictionary. Using lexical affect alone does not capture nuances of spontaneous speech or the semantics of word usage. Instead of completely relying on affect-based lexicons, I propose to use automatic, real-time monitoring of emotion from the signal.

2.5 Summary

Research in language modeling has looked into using prosody as an alternate source of information. However, these methods look into prosody as surface features (see Figure 1.1) rather than manifestation of deeper cognitive processes, and thus, fail to model cognitive factors that can affect speech production. One such cognitive factor is emotion. Emotion plays an important role in human communication, and speaker's emotional state has been shown to have a positive effect on the performance of speech recognizers. In language modeling, use of emotion is scarce and is limited to using increasing the representation of emotion words.

In this dissertation, I present a way to continuously detect the speaker's emotional state and use it in a language model to better predict the speaker's upcoming words.

Chapter 3

Developing a Model for Emotion Recognition

The first step towards building an affect-aware language model is to develop an emotion recognizer. This chapter describes the process of developing an emotion classifier and evaluating its performance.

3.1 Requirements for a Emotion Recognizer

An emotion recognizer for this research should satisfy the following criteria:

1. It should be able to track continuous, moment-by-moment changes in a speaker's emotional state.

2. It should be able to sense subtle changes in a speaker's emotional state.

3. It should be speaker–independent.

An observation fundamental to this dissertation is that the speaker's emotional state can vary over the utterance. For example, the speaker might start his/her utterance on a dominating (high power) note but might end on submissive (low power) cue. Therefore, it is critical for the emotion recognizer to keep a continuous track of the speaker's emotional state so that a word's typical affective-context of occurrence can be modeled. Nicolaou *et al.* [30] used multimodal signals, such as facial expressions, shoulder gestures and audio cues, to track affect continuously. However, in this research, I will use only prosodic cues derived from the speech signal to track affect.

There are several data sources that have been used for emotion recognition from speech [46] [11]. However, these sources are based on acted speech. Developing an emotion recognizer on such sources would not be suitable because its final intended use is for spontaneous speech. Emotion in spontaneous speech is much more subtle than in enacted speech [39]. These subtle changes in variation are important as they could possibly indicate dialog dynamics that would prove helpful for language modeling.

Therefore, I decided to have certain conversations from the Switchboard corpus, a collection of telephonic conversation between mostly unacquainted adults [15], be labeled for affect.

3.2 Labeling Emotion

As illustrated in chapter 2, research in the field of emotion recognition has focused primarily on utterance-level recognition. This research requires moment-by-moment emotional information. Therefore, in this research it was necessary to use data hand-labeled for emotion at sub-utterance level. This section describes the labeling process and presents related observations.

3.2.1 ISG Emotion Annotations

For the purpose of this research, two labelers independently labeled approximately 11.5 minutes of conversations from the Switchboard corpus at the sub-utterance level for perceived activation, valence and power. As a precaution, no track from these conversations contributes data towards the language model.

The labelers used *Dede* for listening to the audio. They selected segments of speech for which they could judge the labels. These segments are called “regions of interest.” Unlabeled segments mostly included regions of silence or regions where the labeler could not judge the label. The labelers were allowed to listen to any amount of context, from past and future, including interlocutor speech. However, the particular version of *Dede* used provided a control for listening to 1.5 seconds of past context from the speaker. For each region of interest, the labelers were asked to provide ratings for activation (A), valence (V) an power (P) independently. Each dimension was rated on an integer scale of 1 to 7, with 7 indicating higher perceived level and 1 indicating lower perceived level. Table 3.1 contains the descriptions given to the labelers for each of the three dimensions. These descriptions of dimensions of emotions are consistent with those given in [1].

Table 3.1: Descriptions for dimensions of emotion that were provided to the labelers.

Dimension	**Description Provided**
Activation	Speaker is active, it sounds like he/she is engaged in the conversation, and is ready to take part in the conversation.
Valence	Speaker's valence is "positive" if he/she sounds upbeat or enthusiastic. A "negative" sound would seem unpleasant and down.
Power	A speaker would sound dominant if he/she is taking control of the conversation or is confident about what is being said.

The labelers were free to select and alter segment boundaries. Additionally, they could also alter their labels, in case they decided they had made a mistake.

3.2.2 Analysis of Emotion Annotations

Since the labelers were free to choose the regions of interest for annotation, many instances lacked annotation from at least one labeler. However, the regions annotated by both lablers accounted for 220 seconds of audio. The unlabeled audio comprised silence or at least missed annotation from one of the labelers. Tables 3.2, 3.3 and 3.4 show the inter-rater agreement matrix for activation, valence and power respectively. Table 3.5 shows the

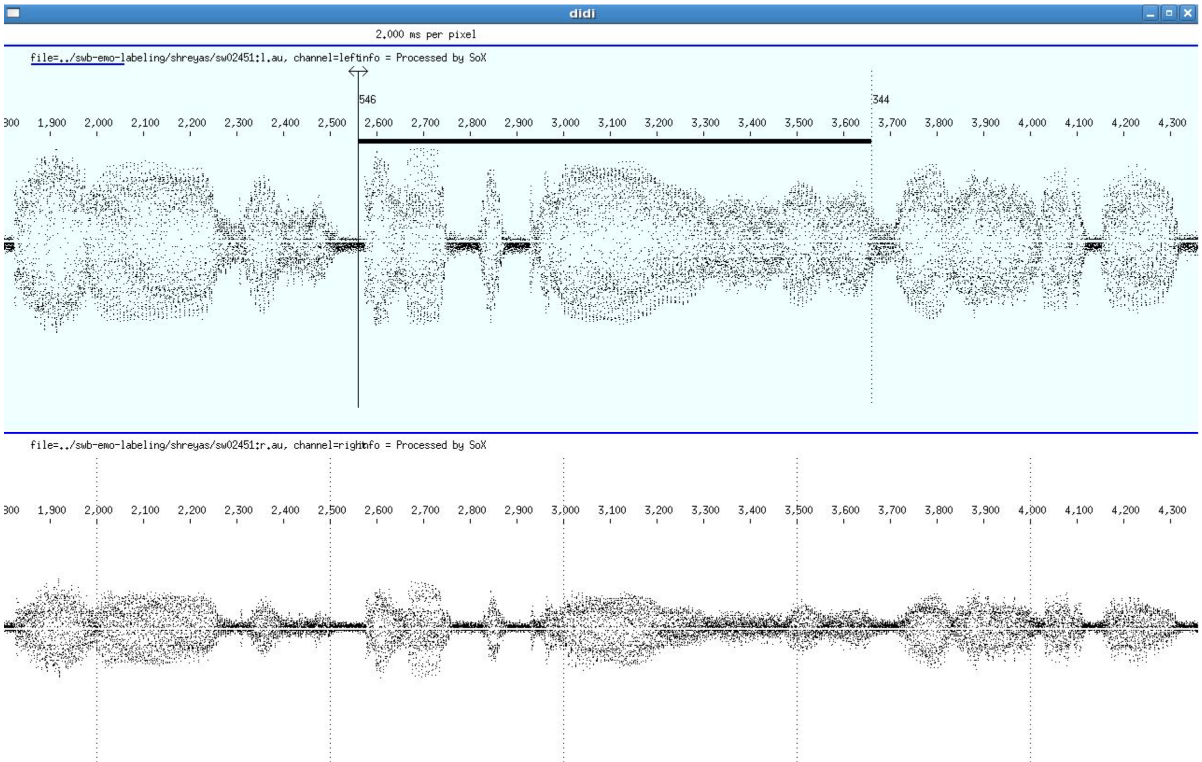

Figure 3.1: Screen shot of labeling with Dede.

quadratic-weight κ coefficient [14] and Pearson's correlation coefficient (ρ), for activation, valence and power, over these regions.

3.2.3 Distributions of and Disagreements over Emotion Labels

Figures 3.2.2, 3.2.2 and 3.2.2 show the cumulative duration of speech plotted against the given label for Activation, Valence and Power respectively. The following observations can be made from these figures:

1. Activation labels from L_1 have a uni-modal distribution with the majority of the regions being labeled "neutral" (4) or "slightly active" (5). However, activation labels from L_2 show a bi-modal distribution with majority of the regions being labeled as "slightly passive" (3) or "moderately active" (6).

Table 3.2: Confusion matrix for Activation labels from L_1 and L_2. Each (x,y) cell corresponds to duration of speech (rounded to the nearest second) that was labeled with activation-level x by L_1 and y by L_2.

		L_1							
		1	**2**	**3**	**4**	**5**	**6**	**7**	**row sum**
L_2	**1**	0	0	0	0	0	0	0	0
	2	0	0	2	5	7	0	0	14
	3	0	1	4	24	24	3	0	56
	4	0	0	2	6	18	2	0	28
	5	0	0	3	14	31	0	0	48
	6	0	0	1	17	37	9	0	64
	7	0	0	0	2	7	2	0	11
	column sum	0	1	12	68	124	16	0	220

2. Both labelers perceived a majority of the regions as “neutral” (4) in terms of valence. However, L_1 perceived many regions as “slightly negative” (3) whereas L_2 perceived many regions as “slightly positive” (5).

3. Power labels for L_2 are more spread out than for L_1. L_1 perceived a majority of the regions as “slightly dominant” (5).

Figures 3.2.3, 3.2.3 and 3.2.3 show the labels for activation, valence and power for the first 10 seconds from the left track of audio 2451.

Table 3.5 represents a high degree of disagreement between the two labelers (L_1 and L_2), especially in the interpretation of power. To uncover possible reasons for disagreement, the two labelers together analyzed the regions where they disagreed the most. The disagreement was measured as the sum of squared difference in terms of the three dimensions (See Equation 3.1).

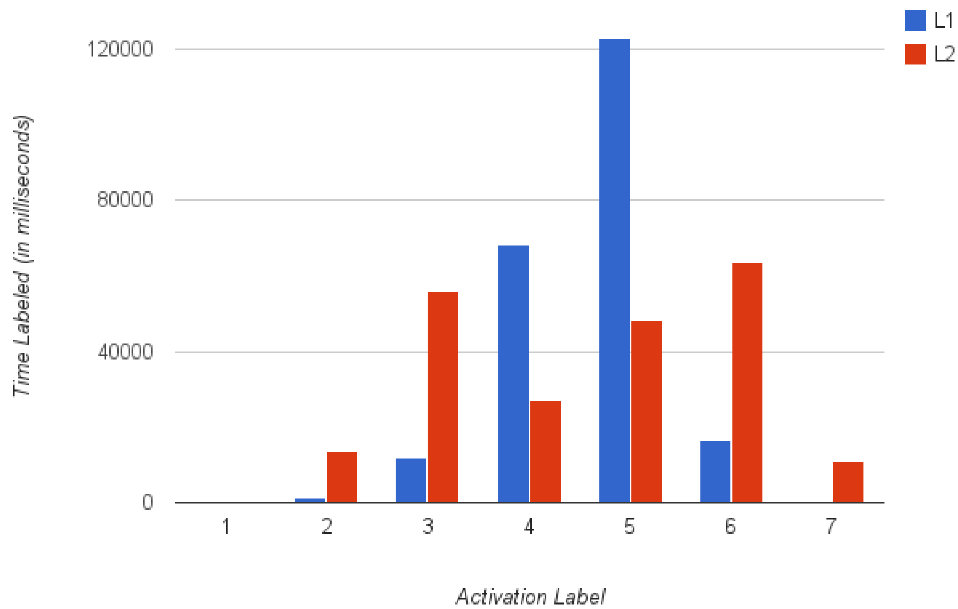

Figure 3.2: Distribution of labels for Activation. X-axis shows the labels for Activation and Y-axis shows the cumulative duration of speech with that label.

$$\triangle = \sum_{X \in A,V,P} (X_1 - X_2)^2 \tag{3.1}$$

The causes of differences included:

1. L_1 and L_2 showed differences in labeling regions where the two participants in dialog produced overlapping utterances.

2. For power, L_1 focused on the underlying cognitive processes involved in turn-taking between the speakers while L_2 looked at the surface-level semantics of the exchange. For example, in audio 2451 at around 45.5 seconds, speaker 'A' takes up the turn

Table 3.3: Confusion matrix for Valence labels from L_1 and L_2. Each (x,y) cell corresponds to duration of speech (rounded to the nearest second) that was labeled with activation-level x by L_1 and y by L_2.

		L_1							
		1	**2**	**3**	**4**	**5**	**6**	**7**	**row sum**
L_2	**1**	0	0	0	0	0	0	0	0
	2	0	0	4	0	0	0	0	4
	3	0	1	11	13	0	0	0	25
	4	0	2	29	71	8	0	0	110
	5	0	3	16	35	7	0	0	61
	6	0	0	0	12	5	1	0	18
	7	0	0	1	0	0	0	0	1
	sum column	0	6	61	131	20	1	0	220

from ‘B’ after ‘B’ has kept the turn for around 30 seconds. This turn is interpreted by L_2 as a low-power turn because of low energy at the start of turn. Whereas L_1 interpreted it otherwise.

3. On several occasions, the disagreement occurred at the start of the conversation. Such disagreement is caused by lack of context at the beginning of the conversation.

4. L_1 was more sensitive towards voicing quality and was less sensitive to local, surface-level contrasts. For example, L_1 interpreted creaky voice as high power, while L_2 interpreted creaky voice as low power.

5. L_2 interpreted sloppy pronunciation to be indicative of lowness in activation and power.

After the analysis, both labelers discussed the source of information on which they focused while labeling. While L_2 paid more attention to surface variations of speech, L_1

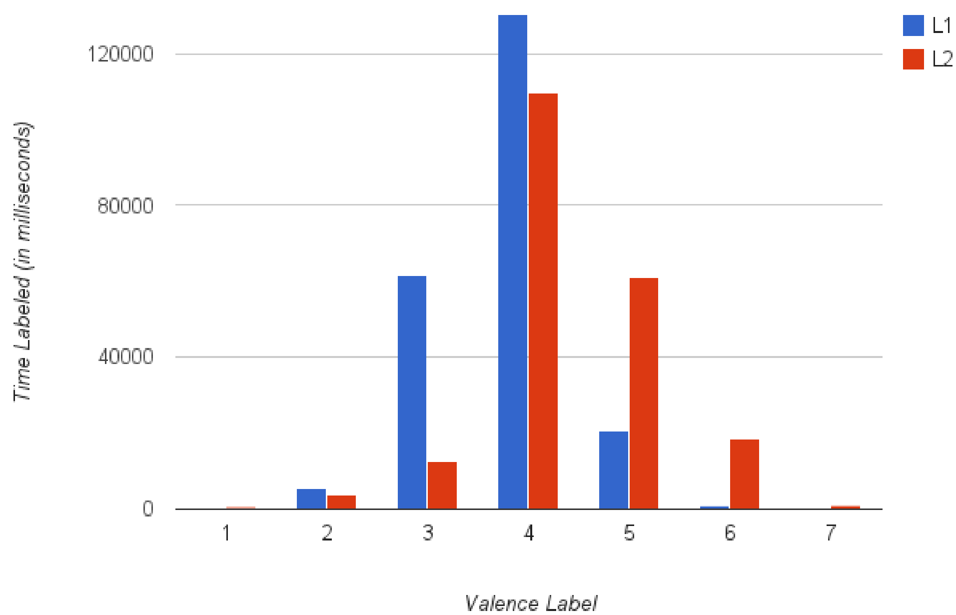

Figure 3.3: Distribution of labels for Valence. X-axis shows the labels for Valence and Y-axis shows the cumulative duration of speech with that label.

was more sensitive to cognitive and psychological changes.

Another factor that might have led to these differences would be include cultural differences in interpreting emotion. In particular, L_1 is a native speaker of American English while L_2 is a non-native speaker. Additionally, the labelers could have been affected subconsciously by their own emotional state.

3.2.4 Annotations Used for Emotion Recognition

Labeler L_2 annotated 20 minutes of conversations, and L_1 annotated 11.5 minutes of conversation. In this dissertation, I have used annotations from L_2 to develop emotion recognizers, because he provided more data. Table 3.6 describes the conversations annotated by L_2.

Table 3.4: Confusion matrix for Power labels from L_1 and L_2. Each (x,y) cell corresponds to duration of speech (rounded to the nearest second) that was labeled with activation-level x by L_1 and y by L_2.

		L_1							
		1	**2**	**3**	**4**	**5**	**6**	**7**	**row sum**
L_2	**1**	0	0	0	0	0	0	0	0
	2	0	1	3	3	9	0	2	18
	3	0	0	6	10	28	10	0	54
	4	0	0	2	14	22	12	0	50
	5	0	0	2	7	22	4	1	36
	6	0	1	2	5	37	10	1	56
	7	0	1	1	0	3	1	0	6
	column sum	0	3	16	39	121	37	4	220

Table 3.5: Inter-rater agreement over the dimensions of emotion.

Dimension	Quadratic-weight κ	Pearson's ρ
Activation	0.206	0.255
Valence	0.211	0.270
Power	0.104	0.124

Table 3.6: Switchboard meta-data of the conversations labeled for emotion.

Conversation ID	**Topic**	**Duration (minutes)**
2451	Soviet Union	10
4319	Job benefits	5
4324	Taxes	5

3.3 Vocal Features for Emotion Recognition

Research on emotion recognition has uses a number of different acoustic features for affect modeling. For example, openSMILE [13] used over 900 prosodic features, Shikler [39] used

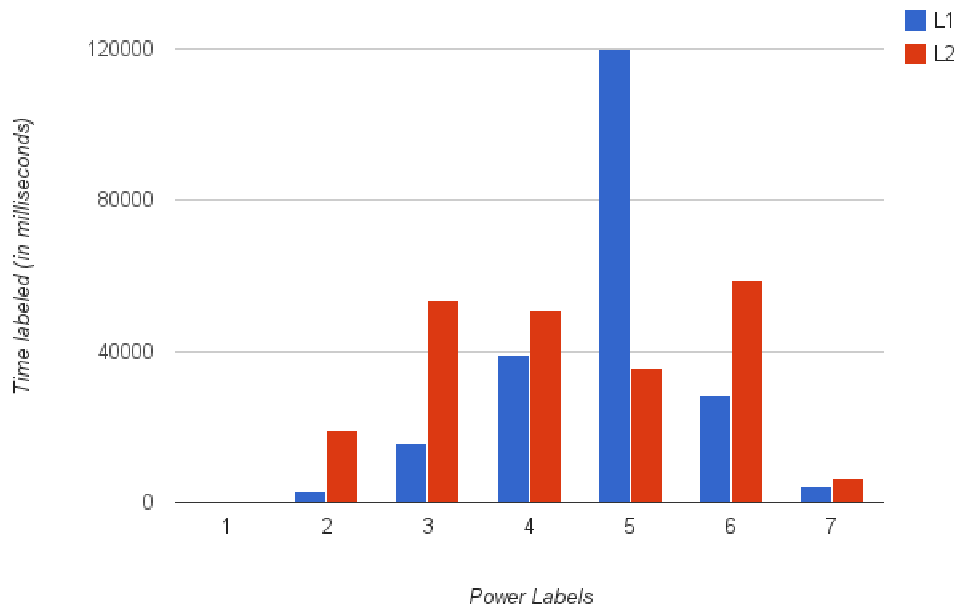

Figure 3.4: Distribution of labels for Power. X-axis shows the labels for Power and Y-axis shows the cumulative duration of speech with that label.

176 while Acosta [1] used 32. A vector of 2000 features was introduced as standard in the first International Audio/Visual Challenge [38]. Moreover, there is ongoing research towards investigating new features that might have value for emotion recognition [21] [27]. In general, the prosodic features used for emotion recognition are related to energy, pitch and duration. For this research, I extracted acoustic features per track over several window sizes using the *respond* module of the Aizula suite and Praat [5]. All together, I extracted 864 features from the audio at 10ms intervals. This section describes the 27 basic features from which these are derived.

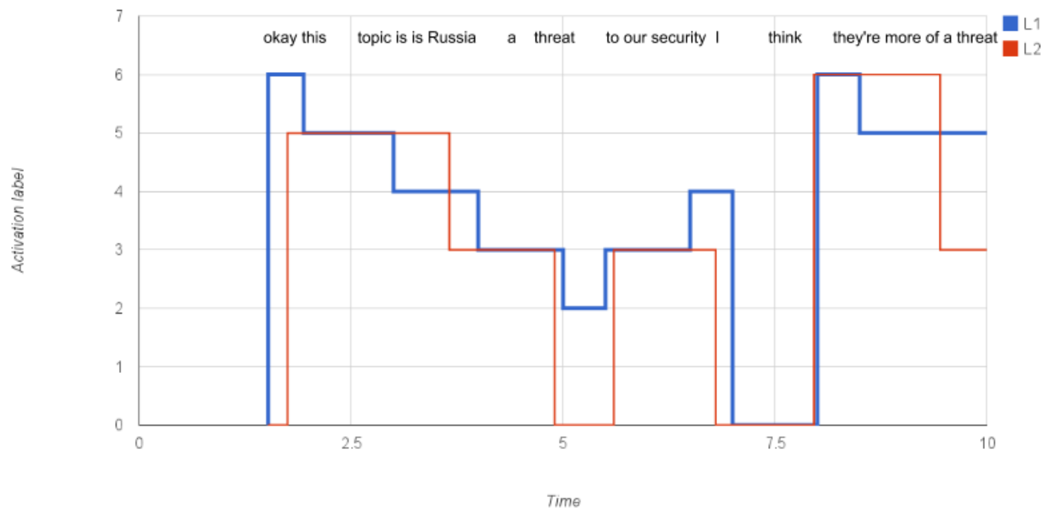

Figure 3.5: Changes in *activation* labels from the two labelers for a segment of audio. Word alignments are approximate.

3.3.1 Description of Features

Praat Features

Praat provides several features to be derived from audio. In this dissertation, I used the following features:

- **Energy** : The energy of a given window is given by the square of its amplitude (A). Over a window containing n samples, the energy is computed as the sum of individual samples as:

$$Energy_{window} = \sum_{i=1}^{n} A_i^2 \tag{3.2}$$

- **Pitch Features** : I extracted several pitch-related features using pitch analysis of sound using Praat. All pitch-related features are expressed in Hertz (Hz). The features are as follows:

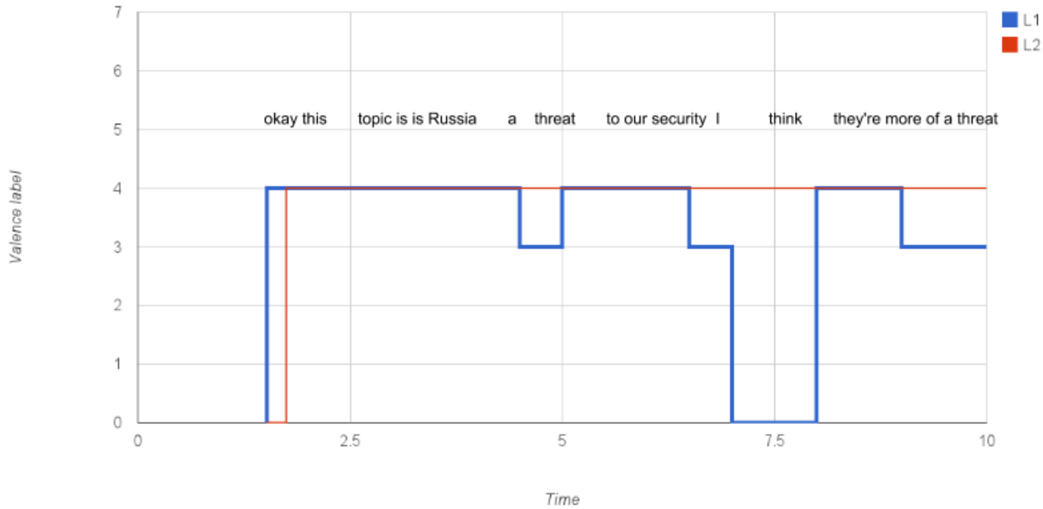

Figure 3.6: Changes in *valence* labels from the two labelers for a segment of audio. Word alignments are approximate.

- **Min Pitch** : The minimum pitch frequency over a given time interval.
- **Max Pitch** : The maximum pitch frequency over a given time interval.
- **Pitch Range** : The difference between the maximum and minimum frequencies that occurred over a given time interval.
- **Median Pitch** : The median of frequencies that occurred over a given time interval.
- **Mean Pitch** : The average of frequencies occurring over a given time interval.
- **SD Pitch** : The standard deviation in the pitch points observed over a given time interval.

• **Jitter Features** : Jitter is a measurement of change in periodicity in a sound sample. Praat provides several metrics to measure Jitter. I extracted the following:

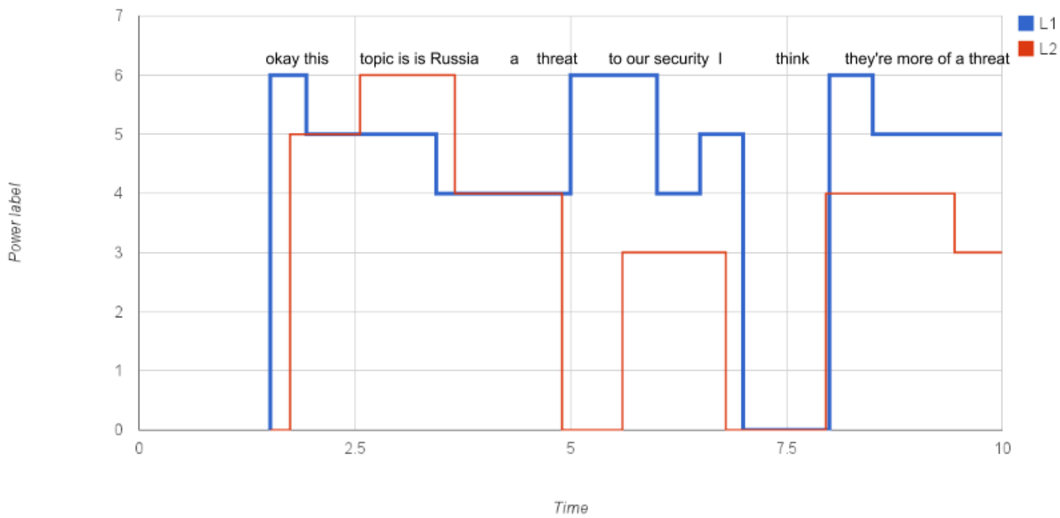

Figure 3.7: Changes in *power* labels from the two labelers for a segment of audio. Word alignments are approximate.

- **Jitter (local)** : The ratio between (a) average absolute difference between consecutive glottal periods, and (b) the average glottal period.

$$Jitter(local) = \frac{\mu_i(|T_i - T_{i-1}|)}{\mu(T)} \quad (3.3)$$

  Here and in subsequent equations, μ_i denotes frame-by-frame average computed over frames in a given window.

- **Jitter (local, absolute)** : The average absolute difference between consecutive glottal periods.

$$Jitter(local, absolute) = \mu_i(|T_i - T_{i-1}|) \quad (3.4)$$

- **Jitter (RAP)** : The ratio between (a) the average absolute difference between a glottal period and the average of it and its two neighbors, and (b) the average glottal period. RAP stands for Relative Absolute Perturbation.

$$Jitter(RAP) = \frac{\mu_i(|T_i - \mu(T_{i-1}, T_i, T_{i+1})|)}{\mu(T)} \tag{3.5}$$

- **Jitter (PPQ5)** : The ratio between (a) the average absolute difference between a glottal period and the average of it and its four neighbors, and (b) the average glottal period. PPQ stands for Period Perturbation Quotient.

$$Jitter(PPQ5) = \frac{\mu_i(|T_i - \mu(T_{i-2}, T_{i-1}, T_i, T_{i+1}, T_{i+2})|)}{\mu(T)} \tag{3.6}$$

- **Shimmer Features**: Shimmer is a measurement of variation in amplitude of sound in a sample. Praat provides several metrics to measure Shimmer. Using Praat, I extracted the following:

  - **Shimmer (loc)** : The ratio between (a) the average absolute difference between the amplitudes of consecutive periods, and (b) the average amplitude.

$$Shimmer(loc) = \frac{\mu_i(|A_i - A_{i-1}|)}{\mu(A)} \tag{3.7}$$

  - **Shimmer (loc, dB)** : This is decibel representation of the average absolute difference between the amplitudes of consecutive periods.

$$Shimmer(loc, dB) = 20 * log_{10}\left(\frac{\mu_i(|A_i - A_{i-1}|)}{\mu(A)}\right) \tag{3.8}$$

  - **Shimmer (APQ3)** : The ratio between (a) the average absolute difference between the amplitude of a period and the average of the amplitudes of it and its neighbors, and (b) the average amplitude. APQ stands for Amplitude Perturbation Quotient.

$$Shimmer(APQ3) = \frac{\mu_i(|A_i - \mu(A_{i-1}, A_i, A_{i+1})|)}{\mu(A)} \tag{3.9}$$

  - **Shimmer (APQ5)** : The ratio between (a) the average absolute difference between the amplitude of a period and the average of the amplitudes of it and four of its neighbors, and (b) the average amplitude.

$$Shimmer(APQ5) = \frac{\mu_i(|A_i - \mu(A_{i-2}, A_{i-1}, A_i, A_{i+1}, A_{i+2})|)}{\mu(A)} \tag{3.10}$$

– **Shimmer (APQ11)** : The ratio between (a) the average absolute difference between the amplitude of a period and the average of the amplitudes of it and ten of its neighbors, and (b) the average amplitude.

$$Shimmer(APQ11) = \frac{\mu_i(|A_i - \mu(A_{i-5}, ..., A_{i+5})|)}{\mu(A)} \tag{3.11}$$

- **NHR** : NHR (Noise–to–Harmonics Ratio) is the ratio between the energy in aperiodic (noise) component of the speech signal and the energy in fundamental and the harmonics of the speech signal.

$$NHR = \frac{fraction\ of\ noise\ samples\ in\ window}{fraction\ of\ voiced\ samples\ in\ window} \tag{3.12}$$

- **MFCC** : Mel–Frequency Cepstrum (MFC) is the representation of the mapping of the power spectrum of audio onto the mel–frequency scale. The Mel–Frequency Cepstrum Coefficients (MFCCs) collectively make up MFC. For this research, I derived six MFC coefficients computed over 40ms wide windows. I used the same window size as mentioned by Nicolaou *et al.* [30].

Respond Features

The Respond module of Aizula suite provides the following acoustic features:

- **Volume** : Estimate of energy in the voice signal.

$$Volume = \frac{E - E_{silence}}{E_{speech} - E_{silence}} \tag{3.13}$$

Here, E is the average per-frame energy of the signal in a window, $E_{silence}$ is the average energy of the signal in no-speech frames over the entire track and E_{speech} is the average energy of the speech frames over the entire track.

- **Pitch Height** : Estimate of the median pitch in the voice signal.

$$PitchHeight = \frac{Pitch_{median} - Pitch_{30\%}}{Pitch_{70\%} - Pitch_{30\%}} + 5 \tag{3.14}$$

Here $Pitch_{median}$ is the median pitch over the window, $Pitch_{30\%}$ and $Pitch_{70\%}$ are the 30^{th} and 70^{th} percentile pitch values and 5 is a static scaling constant.

- **Pitch Range** : Ratio between the 2^{nd} highest and 2^{nd} lowest pitch values in the window.

$$PitchRange = \frac{Pitch[N-1]}{Pitch[2]} \tag{3.15}$$

Here N represents the number of valid pitch frames in the window, and $Pitch[1..N]$ is the sorted array of pitch points in the window.

- **Speaking Rate** : Estimated as the ratio between (a) The average absolute change in energy frame-by-frame and (b) the difference between mean energy of a voiced region and mean energy of a silent region.

$$SpeakingRate = \frac{\mu_i(|E_i - E_{i-1}|)}{E_{speaking} - E_{silence}} \tag{3.16}$$

3.3.2 Normalization of Speaker-dependent Features

To correct for speaker variability, the features need to be normalized. All Praat-derived features (except MFCCs) are z-normalized per track so that the mean (μ) is 0 and standard deviation (σ) is 1. The z-normalized feature value is given by the formula:

$$x_{znorm} = \frac{x - \mu_x}{\sigma_x} \tag{3.17}$$

In several cases, the feature extractor returns a special token for a feature if that feature cannot be reliably computed over a specified window. These instances are assigned the mean value (=0) after z-normalization. Numeric replacement of unreliable feature value tokens was necessary for me to be able to use them for building recognizers. I chose the value of replacement to be the mean value as a design choice.

MFCCs are left out of the normalization process because of their complex nature – cosine transform coefficients of the real logarithm of the short-term energy spectrum. Normalizing MFCCs might result in loss of information and feature quality itself. In addition, I could not find prior research in emotion recognition that normalized of MFCC-related features [30] [37].

3.4 Development of the Emotion Recognizer

Using the features described in Section 3.3 and the annotations mentioned in Section 3.2.4, I developed and evaluated emotion recognizers for each of the three dimensions. This section presents the method used for developing an emotion recognizer.

Following previous work, in this research I assume that acoustic properties of a voice signal correlate with the emotional state of the speaker. However, no precise models of emotion interpretation yet exist, especially for moment-by-moment emotion tracking. Therefore, using several machine learning algorithms provided by Weka [16], I developed my own models for emotion recognition and evaluated their accuracy. In this dissertation, I have used Weka version 3. 7. 9. to develop different emotion recognizers. This section describes the process of developing models for emotion recognition. Figure 3.8 shows a schematic view of developing emotion classifiers.

3.4.1 Complete Feature Set

Levenson [24] suggests that emotions last for approximately 0.5 to 4 seconds. Keeping this in mind, I extracted the various features mentioned in Section 3.3 over window sizes ranging from 100ms to 2000ms centered at word onset. Figure 3.9 shows a schematic diagram representing prosodic feature context windows.

For initial experimentation, I used features from previous/left context alone. The size of the full feature set for left context alone was 216 (21 features extracted over 10 window sizes plus 6 MFC coefficients). This feature set contains features only from the region marked red in Figure 3.9.

Next, to test whether future/right context plays a role in determining the current emotional state, I augmented the left context features with their right context counterparts.

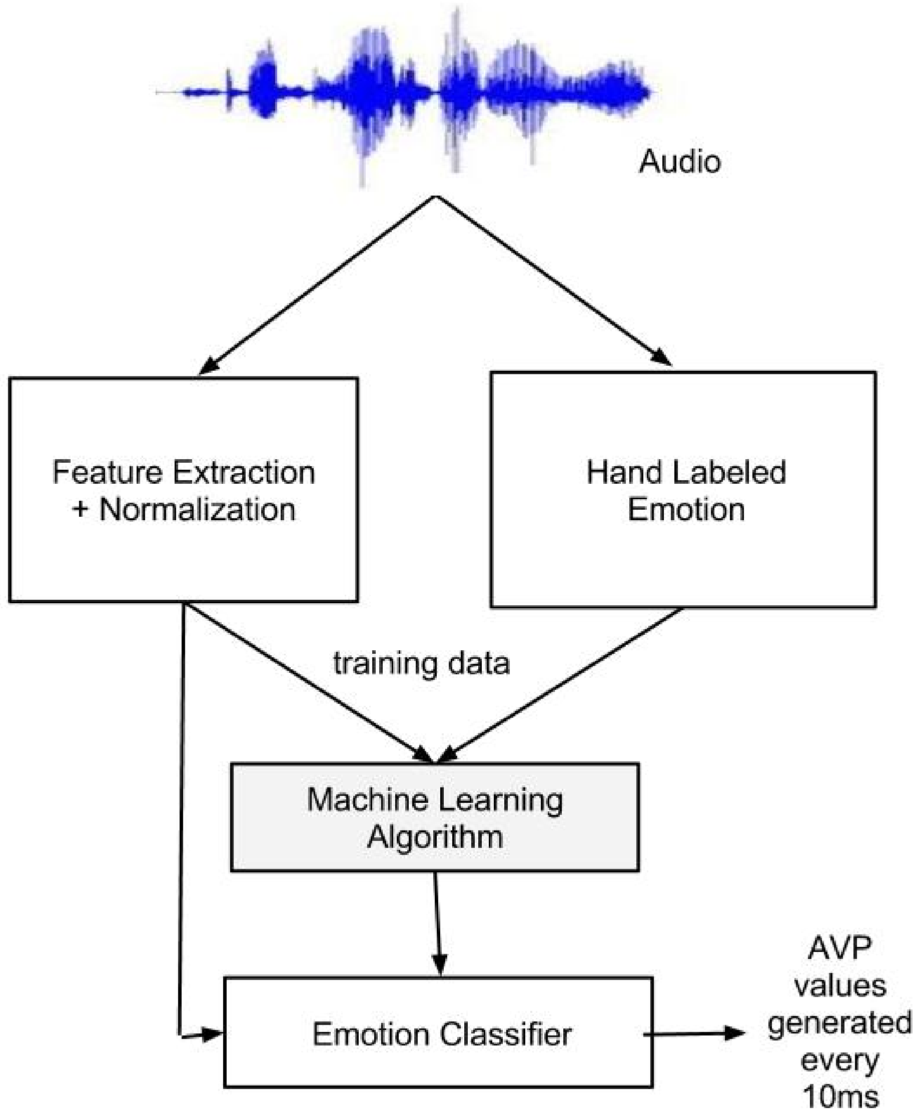

Figure 3.8: Schematic diagram of developing an emotion classifier.

This resulted in 432 features: 216 from previous context and 216 from future. This feature set contains features computed over regions marked red and blue in Figure 3.9.

Finally, I included past and future context features from the interlocutor's track, making the final feature set size 864. This feature set is represents features from the red, blue and green regions in Figure 3.9.

3.4.2 Associating Labels with Features

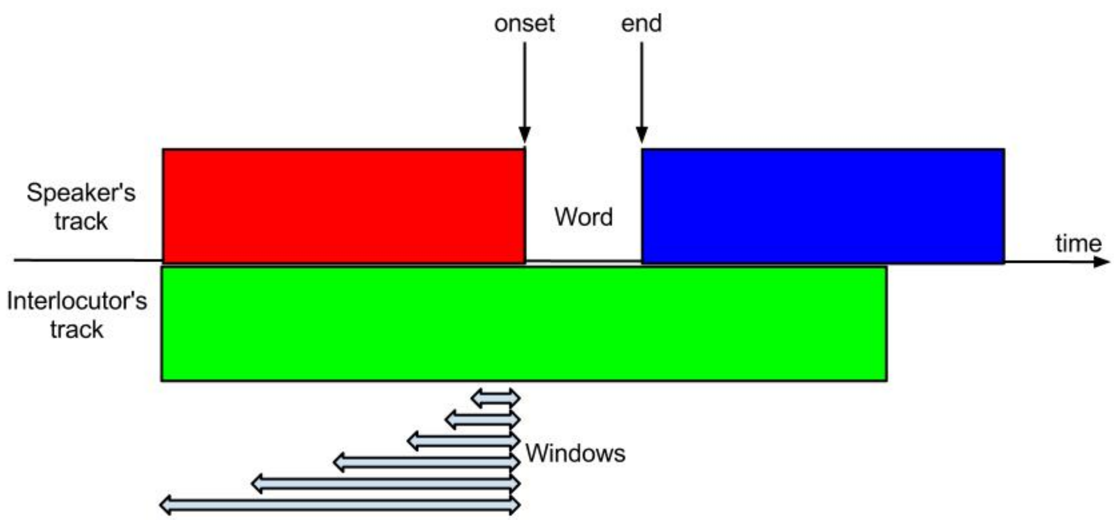

Figure 3.9: Diagram representing contextual prosodic feature windows. The red region represents features computed over different window sizes (up to 2000ms) to the left of the point-of-interest. The blue region represents the same features computed to the right of the point-of-interest. The green region represents these features computed over the interlocutor's track. The word's own prosody is not used.

The critical point for an emotion recognizer's accuracy is the emotion inferred at the word onsets. This is because, based on the recognized emotional state at word onset, the language model will adjust the estimates for different words. Therefore, in this research, I associated emotion labels with word-onset features. Additionally, the word's own prosody is left out while developing models that include future-context, thereby preventing it from affecting the emotion recognizer's performance, because the prosodic characteristics of the word itself are already modeled, to some extent, by the acoustic model of the speech recognizer and, thus, need not be modeled by the language model. Figure 3.9 shows a diagrammatic representation of the word's own prosody being left out.

3.4.3 Developing and Evaluating Models for Emotion Recognition

Associating L_2's emotion labels with the word onset features for the corpus subset he labeled resulted in 3341 word tokens with emotion labels. Out of this set, 281 labels (corresponding to a single randomly selected track) were held out as a test set. I used several machine-learning classification models for recognizing the three dimensions of emotion independently. All models were trained with the remaining 3060 observations as the training set and with the default parameters provided by Weka.

Tables 3.7, 3.8 and 3.9 show the correlation coefficients between the predicted values and the human labels for the test set for activation, valence and power, using (a) speaker's past context only, (b) speaker's past and future context and (c) speaker's and interlocutor's past and future context. Although, many models that were developed, these tables show only the performance of some of the better-performing models. For each column in Tables 3.7, 3.8 and 3.9, values in bold indicate the best performance, in terms of correlation, obtained using different context features.

3.4.4 Analysis of Model Performance

Tables 3.7, 3.8 and 3.9 show that speaker's future prosodic context is helpful in improving the performance of an emotion recognizer for activation and power. However, using information from the interlocutor's track is not beneficial for these dimensions. This can perhaps be attributed to interlocutor's features likely being mostly unreliable when only the speaker is speaking. The unreliability might be caused by line noise and bleeding across tracks. As mentioned in section 3.3.2, in many cases, when a feature cannot be computed reliably, the feature extractor returns a special token that signifies feature unreliability.

Table 3.7: Performance of different models on the test set for the dimension of **activation**, in terms of Pearson's correlation coefficient (ρ), using different context features. '–' indicates low or negative correlation.

Model	S (L only)	S (L & R)	S & I (L & R)
SVM (ϵ-SVR + linear kernel)	0.23	0.23	0.15
SVM (ν-SVR + linear kernel)	0.20	0.22	0.11
Linear regression	0.21	0.29	0.17
M5Rules	0.20	0.21	–
Decision Stump	0.19	0.17	0.18
SMOReg (with normalization)	0.25	0.29	0.16
Gaussian Process	**0.26**	**0.32**	**0.24**
REPTree	0.09	0.07	–
M5P Tree	0.02	0.05	0.03
Multi-layer Perceptron (MLP)	–	0.05	–
MLP Regressor	0.20	0.23	0.23

After z-normalization, these values are assigned the mean value (=0). In the absence of explicit feature selection, it might be that the model gives inappropriate weight to some features. In the case of valence, the interlocutor's context seems to be valuable as the performance improves over models developed using speaker's own context alone.

An important observation can be made from Tables 3.7, 3.8, 3.9 and 3.5: The Pearson's correlation between the models' prediction and the test instances consistently outperform the Pearson's correlation between the two labelers over the annotated conversations. This suggests that the emotion recognizers are at least better at predicting what L_2 might perceive than a human.

The best models I obtained, however, have a much lower correlation than reported in other research. For example, Acosta [1] reported prediction correlations of 0.73, 0.44 and

Table 3.8: Performance of different models on the test set for the dimension of **valence**; as before.

Model	S (L only)	S (L & R)	S & I (L & R)
SVM (ϵ-SVR + linear kernel)	0.02	0.07	–
SVM (ν-SVR + linear kernel)	0.11	0.03	–
Linear regression	0.04	0.06	–
M5Rules	**0.15**	0.06	0.08
Decision Stump	–	0.03	0.02
SMOReg (with normalization)	–	0.08	0.06
Gaussian Process	0.07	0.12	0.09
REPTree	0.11	**0.14**	0.04
M5P Tree	0.11	0.04	–
Multi-layer Perceptron (MLP)	0.14	0.11	**0.19**
MLP Regressor	0.11	0.14	0.04

0.79 for the dimensions of Activation, Valence and Power respectively.

Several factors might explain this. First, the set of training data was small. It is generally possible to achieve a better correlation with more labeled data. Second, these models attempt to recognize word–by–word changes in emotional state, which is much harder than recognizing emotional state conveyed over an entire utterance. Nicolaou *et al.* [30] reported similarly low correlation values for activation and valence (0.419 and 0.146 respectively, using SVMs). Third, the labeled data was biased towards some labels. This is specifically true for the valence dimension, where the label "4" is used for more than 50% of the labels (See Figure 3.2.2). Such bias has been observed in other emotion databases as well [8] [29].

Another statistic used to calculate the goodness of fit is R^2. The R^2 for the best-

Table 3.9: Performance of different models on the test set for the dimension of **power**; as before.

Model	S (L only)	S (L & R)	S & I (L & R)
SVM (ϵ-SVR + linear kernel)	0.09	0.19	0.08
SVM (ν-SVR + linear kernel)	0.11	0.18	0.12
Linear regression	0.12	0.22	0.09
M5Rules	0.1260	–	0.2465
Decision Stump	0.13	0.14	0.14
SMOReg (with normalization)	0.12	0.26	0.17
Gaussian Process	**0.16**	**0.27**	0.19
REPTree	–	0.03	0.01
M5P Tree	0.13	0.03	0.19
Multi-layer Perceptron (MLP)	0.04	0.19	**0.24**
MLP Regressor	0.12	0.09	0.10

performing model for activation was calculated to be 0.10, 0.075 for power and 0.035 for valence.The R^2 metric is good for computing the goodness of fit for linear regression models. However, the models used here are non-linear in nature and therefore this metric does not indicate the true goodness of these models.

For my language modeling method, where I model based on bins of values, it is more important to identify the polarity of speech segments – high, medium or low – rather than predict the exact value on a particular dimension. To measure how well each of the models performs in terms of identifying the polarity of a speech segment, I use the sign agreement metric (SAGR) [30], which is computed using equation 3.18.

$$SAGR = \frac{\sum(\delta(predicted, actual))}{N} \tag{3.18}$$

Here, N represents the size of the test set, and $\delta(a, b)$ is the delta function:

$$\delta(a, b) = \begin{cases} 1, & \text{if } a = b \\ 0, & \text{if } a \neq b \end{cases} \tag{3.19}$$

As mentioned in section 3.2.1, each dimension was annotated on a 1-7 scale. I divided this scale into three segments: [1, 3] as low, (3, 5) as neutral and [5, 7] as high. For computing $SAGR$, the delta-function returns 1 if the predicted and actual values fall in the same segment. Thus, the modified delta-function is given by:

$$\delta(a, b) = \begin{cases} 1, & \text{if } segment(a) = segment(b) \\ 0, & \text{if } segment(a) \neq segment(b) \end{cases} \tag{3.20}$$

Table 3.10 shows the SAGR metric computed for the best performing emotion recognizers and SAGR computed using the most common label in the training data.

Table 3.10: Sign agreement of the best-performing models on Activation, Valence and Power compared with the sign agreement for the dominant label.

Dimension	SAGR	SAGR (dominant label)
Activation	0.39	0.23 (6)
Valence	0.68	0.78 (4)
Power	0.35	0.08 (6)

In terms of activation and power, the values predicted by the emotion recognizer fall into the correct segments more often than by using the most common label. For valence, however, the dominant value classifies values better than the recognizer's prediction. This could be because valence shows little variation in the data (see figure 3.2.2). For power, the labels are more evenly distributed (see figure 3.2.2). Additionally, this could also be due to the fact that there is a large difference in the label distributions between the training and the test sets for valence.

3.5 Summary

A continuous emotion recognizer should be able to track and detect subtle changes in the speaker's emotional state. To develop such a recognizer, 40 minutes of track audio was labeled for activation, valence and power. Using the speech processing packages Praat and Respond, prosodic features over several window sizes were derived. Several machine learning algorithms from Weka were then used to predict the levels of activation, valence and power. For activation and power, speaker's future context provides useful information for predicting current levels of activation and power. For valence, the interlocutor's context is more useful. Although, the correlations between actual and predicted values are low for all three dimensions, the sign agreement metric confirms that these models can frequently detect subtle variations in activation and power.

Chapter 4

Using the Emotion Recognizer in a Language Model

This chapter describes the corpus used in this research and discusses the general strategy for evaluating emotion-based language models. Figure 4.1 shows a schematic diagram of an affect-adaptive speech recognizer. Instead of using raw prosodic features as shown in Figure 2.1, the language model uses the emotional state computed from the raw features using the emotion recognizer for predicting the upcoming word.

4.1 Corpus

In this research, I used the Switchboard corpus of telephonic conversations [15]. For language modeling, a subset of 981 tracks of audio (approx. 80 hrs. of speech, 650K tokens) is used for training. The tuning set consists of 50 tracks of audio (35K tokens). This set is mainly used for feature selection and parameter optimization. The test set consists of 45 tracks of audio (approx. 4 hrs. of speech, 28K tokens). All three sets are mutually exclusive. This corpus is the same as that used in [49] and [45].

4.2 Evaluation Metric

Perplexity is a commonly used metric for evaluating language models: perplexity is the size of an imaginary word list whose words are equally probable [19]. Smaller perplexity values are indicative of better language models. Mathematically, perplexity is computed

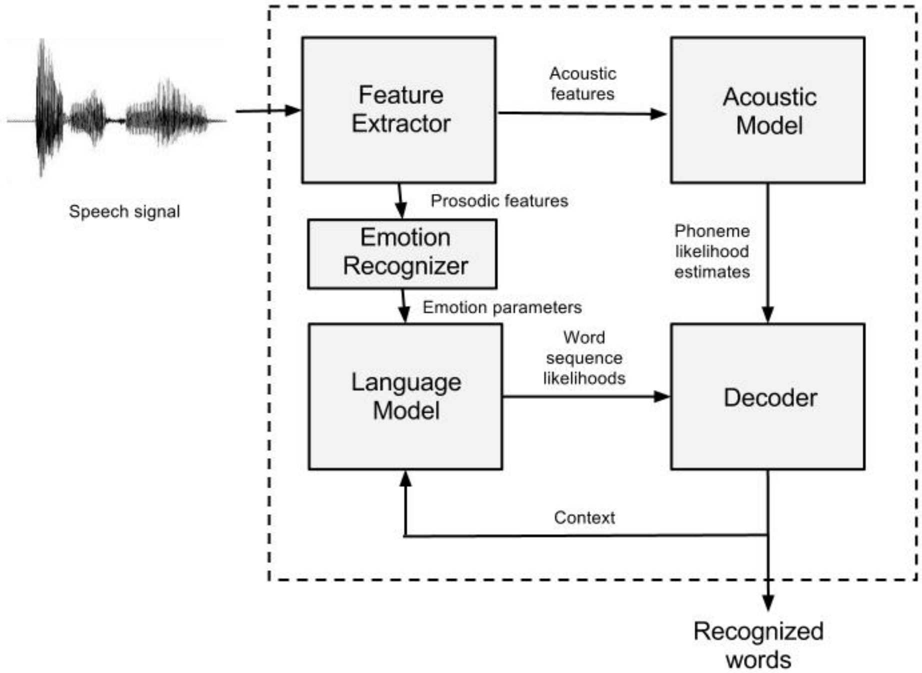

Figure 4.1: Schematic representing a speech recognizer using an affect-adaptive language model.

using equation 4.1, where $P(w_i)$ is the probability estimate for w_i and T is the number of word tokens in the test set.

$$Perplexity = \sqrt[T]{\prod_{i=1}^{T} \frac{1}{P(w_i)}} \tag{4.1}$$

4.2.1 Baseline Performance

I used the back-off trigram model from SRILM Toolkit (version 1.5.6) [43] as the baseline language model. This model has a perplexity of 109.449 on the test set. The vocabulary is limited to the 5000-most frequent words occurring in the training set. All other tokens are treated as out-of-vocabulary(OOV) and are excluded from perplexity computation.

4.3 Binning

As mentioned in section 3.4.4, for my language-modeling purposes, it is more important for the models to detect the difference in polarity than to predict the exact value on a dimension. For this purpose, emotion predictions on each of the dimensions are binned into one of three classes. The class boundaries are same as those used to compute the sign-agreement metric (SAGR) mentioned in section 3.4.4. Each class is assigned a unique context identifier (See Figure 4.2). In addition, a fourth context identifier is used for cases where a recognizer fails to predict values. This label occurs more frequently at the start of a conversation, when few or no features can reliably be computed.

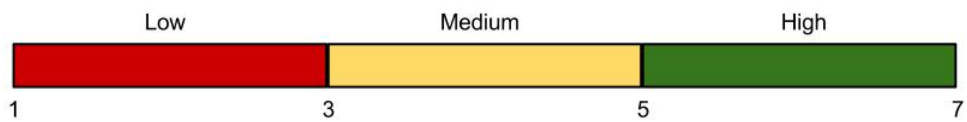

Figure 4.2: Bin thresholds and context identifiers for predicted emotion.

4.4 Combination with N-grams

The method of combining context-based emotion estimates with the n-gram model is identical to the one used in [45]. This section briefly summarizes this process.

After binning, the predicted emotion features at word-onset are filtered using the ISIP transcriptions of the Switchboard corpus [17]. For each word in the vocabulary, I generate its distribution of occurrence over the context identifiers for each emotion dimension. This distribution is converted into R-ratio [49], which is a measure of likelihood of an observation given a context identifier, using equation 4.2.

$$R(w|c) = \frac{P(w|c)}{P(w)} \tag{4.2}$$

Here, $P(w|c)$ is the smoothed probability for the word w occurring in context c and $P(w)$ is the unigram probability for w.

The informativeness of a word's R-ratio in a given context is measured by using the χ^2 test. The confidence (q) is computed using this test. The R-ratio is raised to the q^{th} power, to generate the S-ratios (equation 4.3). This is important in case of infrequent words.

$$S(w|c) = R(w|c)^q \tag{4.3}$$

These S-ratios are applied as scaling factors to the n-gram estimate for the word using equation 4.4.

$$P_{scaled}(w|c) = P_{ngram}(w) \times S(w|c)^k \tag{4.4}$$

Here, k denotes the credence given to a feature in the final combination. Optimal k-values are determined independently using the tuning set.

Finally, the scaled estimates are normalized to give true probability values.

4.5 Parameter Optimization for Individual Dimensions of Emotion and Evaluation on Tuning Set in Isolation

As mentioned in Section 4.4, credence parameters (k) need to be optimized for each dimension. This section illustrates the process of obtaining the optimum parameter values using the tuning set.

4.5.1 Evaluation with Default Parameters

Using the default value (= 0.3) for credence, each dimension of emotion is first evaluated in isolation with the n-gram language model. Table 4.1 shows the perplexity values obtained for each of the three dimensions and the relative improvement over the baseline perplexity for the tuning set, 108.443.

Table 4.1: Perplexity benefit obtained over the baseline using default credence values.

Dimension	**Perplexity**	**Reduction**
Baseline	108.443	–
Activation	108.187	0.23%
Valence	108.439	0%
Power	108.106	0.31%

4.5.2 Optimization of Credence Parameters in Isolation

For each dimension, the credence parameter (k) was then optimized in isolation. Using a step-based approach the credence was changed and the resulting perplexity (over the tuning set) recorded. Table 4.2 shows the optimal credence values obtained for each dimension and the perplexity value obtained. Using activation and power predictions alone shows a minor improvement in performance. However, valence does not show any improvement over the baseline.

Table 4.2: Optimized perplexity values and credence values.

Dimension	**Optimal k**	**Perplexity**	**Reduction**
Activation	0.7	108.067	0.35%
Valence	0.3	108.439	0%
Power	0.9	107.835	0.56%

4.6 Improving the Bin Thresholds

The binning thresholds mentioned in section 4.3 were determined *a priori*. The idea behind selecting these values was to divide the seven-point scale into three equal segments. In ideal conditions, the distribution of data would be almost equal across these segments. However, most of the predictions for the training data (approx. 85%) fell in the *medium*-bin (see Figures 4.3, 4.4 and 4.5). This weakens the models' ability to provide valuable information, as most of the data fell into a single bin.

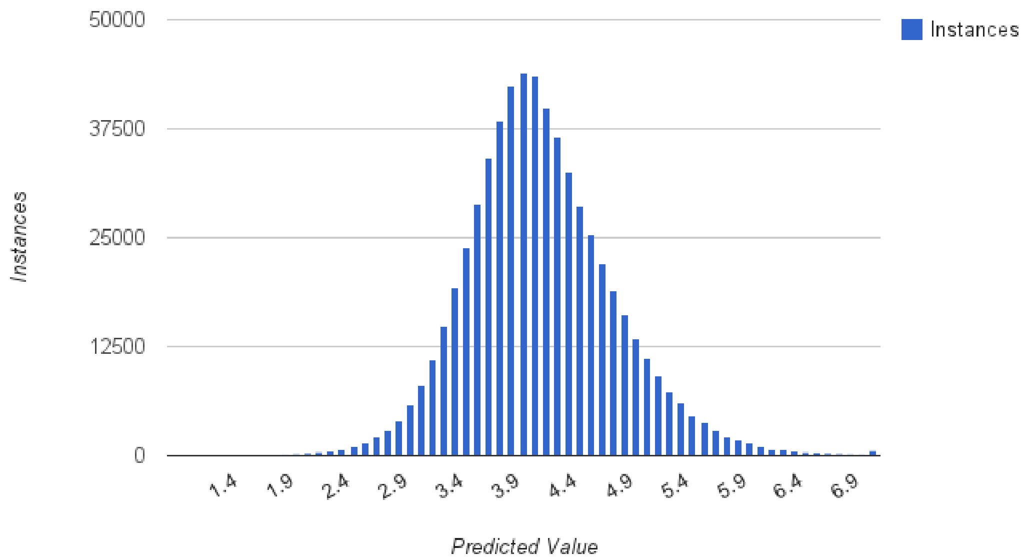

Figure 4.3: Graph representing the distribution of predicted *activation* labels in the training set.

To overcome this weakness, I analyzed the predictions on each emotion dimensions and chose new thresholds for the bins. I chose the new *low-medium* as 3.5 and the new *medium-high* threshold as 4.5 for power and activation because they are close to the 25^{th} and 75^{th} percentile values for the three dimensions. For valence, I used a different binning strategy. Observing that the model uses a single value for most of the predictions, I classified values

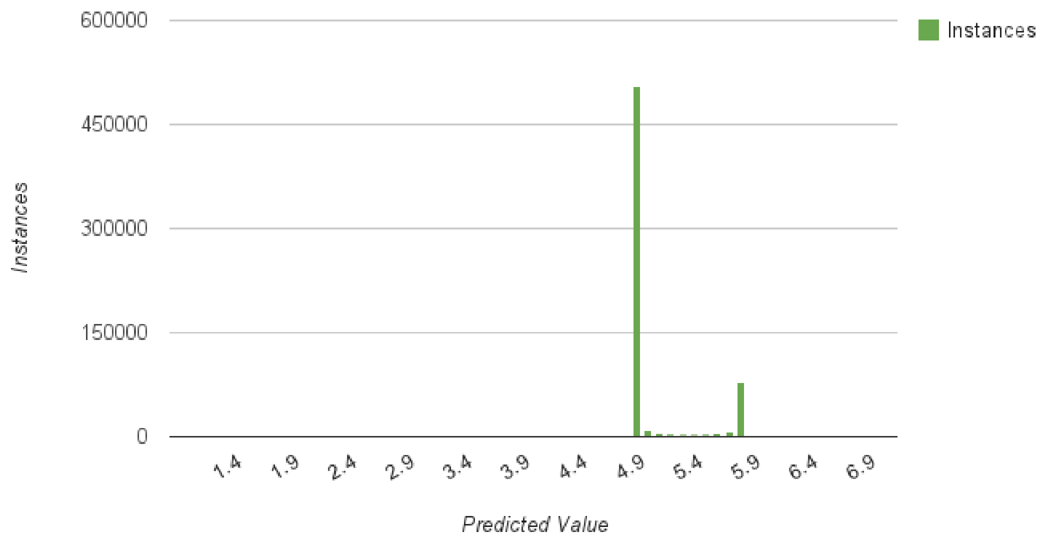

Figure 4.4: Graph representing the distribution of predicted *valence* labels in the training set.

lower than that as *low* and values higher than that as *high*. The rest of the predictions are classified as *medium*.

Table 4.3 shows the perplexity values obtained over the tuning set after binning predictions using the new thresholds and re-optimizing the credence parameter (k). The improvement in performance over the baseline shows the effect of using proper thresholds for binning. Figure 4.6 shows the perplexity benefit obtained as function of the credence parameters for activation, valence and power, each evaluated in isolation.

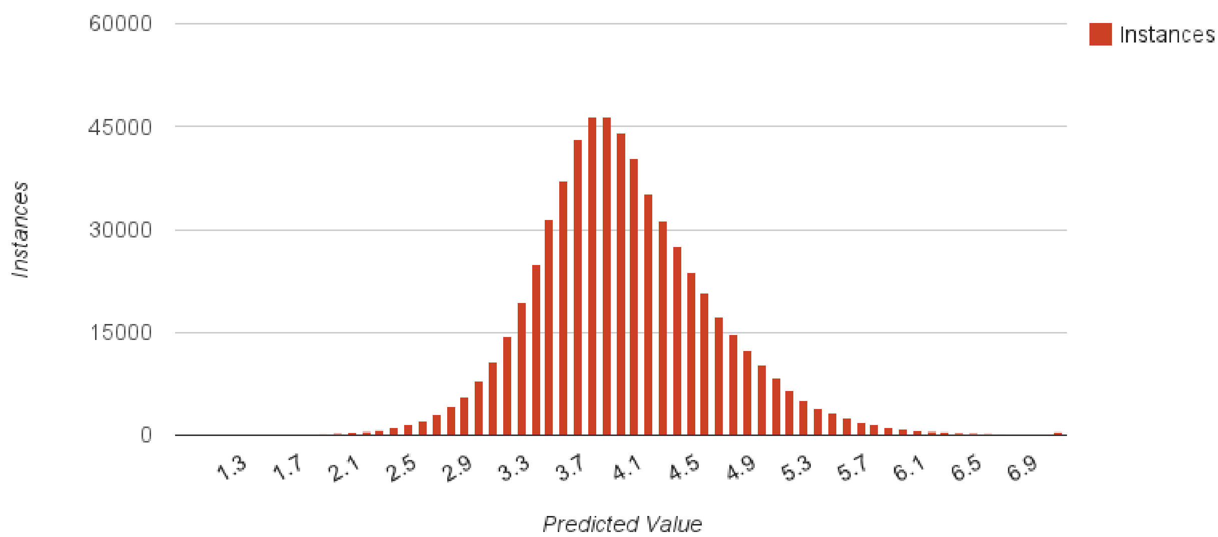

Figure 4.5: Graph representing the distribution of predicted *power* labels in the training set.

Table 4.3: Optimized perplexity values and credence values when evaluating the dimensions in isolation.

Dimension	**Optimal** k	**Perplexity**	**Reduction**
Baseline	–	108.443	–
Activation	0.70	107.676	0.71%
Valence	0.65	108.002	0.41%
Power	0.70	107.464	0.90%

4.7 Optimization of Credence Parameters in Combination

This section presents the results obtained after tuning the emotion-based language model with all three dimensions tuned in combination.

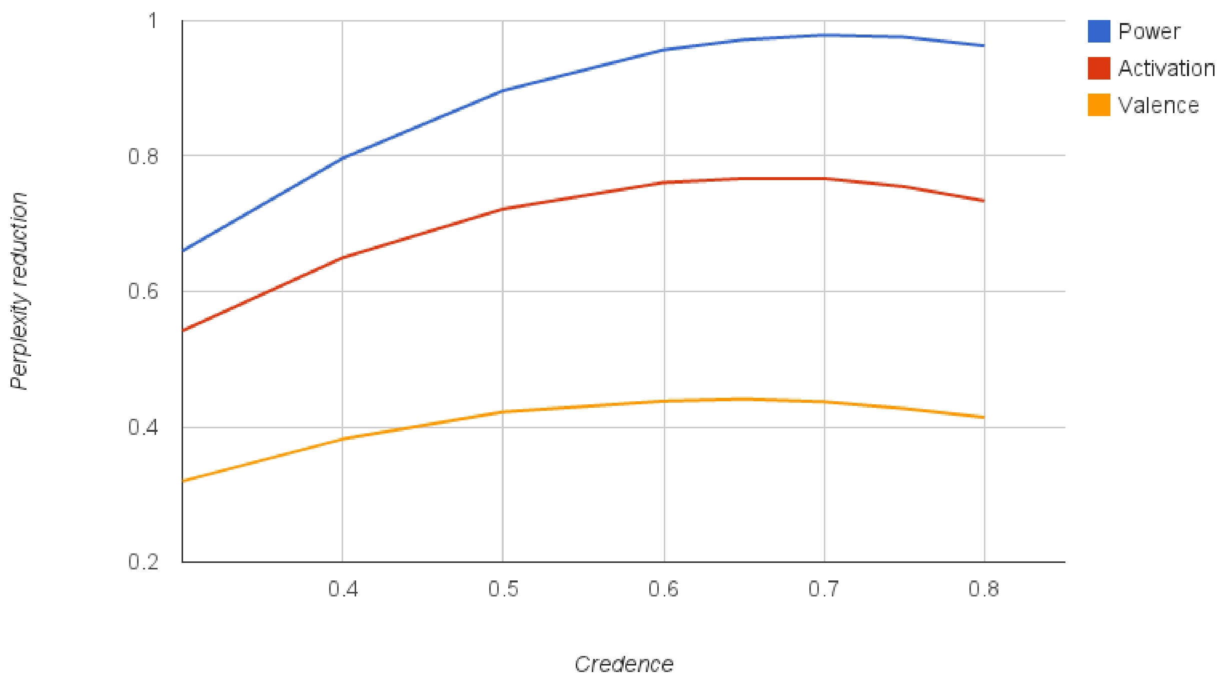

Figure 4.6: Graph representing the perplexity benefit obtained as a function of credence k for activation, valence and power in isolation.

Ward *et al.* [49] mentioned that simple multiplication of the scaling factors from different dimensions in Equation 4.4 can cause multiple redundant penalties to hurt the final estimate for a word token. Thus, the credence parameters for the different dimensions need to be re-evaluated in combination. Rather than using hill-climbing, I used the direct approach of evaluating all combinations for credence parameters; this was possible because of the small number of dimensions.

Table 4.4 shows the optimal credence values obtained in combination for the dimensions of emotion. The optimal perplexity for these credences was 107.146 – a 1.2% reduction over the baseline perplexity of 108.443.

Table 4.4: Optimized credence values for the dimensions when evaluated in combination.

Dimension	**Optimal k**
Activation	0.30
Valence	0.50
Power	0.45

4.8 Results

Once the credence parameters were optimized using the tuning set, they were used to evaluate the model on a test set. This section presents the perplexity results obtained on the test set.

Using the optimal k values from tables 4.3 and 4.4, I evaluated the benefit of using each dimension in a language model, first in isolation and then in combination on the test set. Table 4.5 shows the perplexity values obtained on the test set.

Table 4.5: Perplexity values obtained on the test set using optimized credence parameters.

Dimension	**Perplexity**	**Reduction**
Baseline	109.449	–
Activation (A)	108.580	0.79%
Valence (V)	108.972	0.44%
Power (P)	108.295	1.05%
A + V + P	**107.970**	**1.35%**

In isolation, power showed the best improvement in perplexity followed by activation and valence. When used in combination, these dimensions give a 1.35% improvement in LM performance over the baseline.

4.9 Summary and Significance

The predictions generated by the emotion recognizers are binned depending upon how high or how low they are. For each word, the fraction of its tokens that fell in each bin for each dimension is computed. This fraction is raised to a power, k, called the credence parameter and is used as static multiplicative factor to the trigram probability estimate. The k-value is computed independently for each dimension and used in the final evaluation.

Using ad-hoc binning method resulted in poor performance of the models. The distributions of raw values were analyzed and the binning thresholds were adjusted based on the analysis. This improved the performance of the language models.

Using all dimensions in combination for language modeling reduced the perplexity by 1.35%. This shows that inferring the speaker's current emotional state can help predict the speaker's next word. However, the improvement is roughly half of that obtained using prosody alone. Using simple combinations of basic prosodic features (volume, pitch height, pitch range and speaking rate) computed over windows at word onset, Ward *et al.* [49] reported a perplexity improvement of 2.6% over the baseline tri-gram backoff model.

Chapter 5

Discussion and Analysis

This chapter presents an analysis of the predictions generated by the emotion recognizer and discusses the results obtained by using an emotion recognizer in a language model.

5.1 Performance of the Affect-Adaptive Language Model

The results shown in Table 4.5 indicate that information on the speaker's current emotional state can improve the performance of a language model. In particular, power provides the best improvement. This might be attributed to patterns of turn-taking in spontaneous dialog. That is, the speaker might start off on a dominant note, which is typical of turn-grab, and might end on a submissive tone signaling a turn-yield. Although, I did not specifically test or model turn-taking patterns, my models might be capturing this based on prosodic information.

Of the three dimensions, valence gives the least improvement. In a corpus like Switchboard, it is hard to find large variations in valence. Most of the conversations are friendly and are either neutral or slightly positive. Such a bias makes it difficult to build models that are sensitive to these variations. Additionally, valence is hard to compute using acoustic cues alone [30].

The performance improvements obtained support the first claim mentioned in Section 1.2: using an emotion recognizer can reduce the perplexity of a language model. However, the perplexity reduction obtained is not sufficient to prove the the second claim: using an emotion recognizer would perform better than raw prosody in a language model.

5.2 Analysis of Tendencies for Words to More Certain Emotions

I analyzed the predictions generated by the models for words in the language model's vocabulary. In particular, I analyzed the fraction of occurrence for each word for a given context-identifier. For each context-identifier, the words are sorted based on the fraction of their occurrences for that context-identifier. From these sorted lists, Tables 5.1, 5.2 and 5.3 show the top 25 words for each context-identifier on activation, valence and power dimensions. These tables show only words that occur at least 100 times in the training corpus.

Backchannels such as *uh*, *hum*, *yeah* etc. seem to be more common in the low activation and low power regions than anywhere else. This seems intuitive because backchannels are single-word utterances that are often preceded and followed by silent regions. Because both activation and power are modeled from speaker's past and future context, these results are expected. For regions of high power and high activation, contractions and words containing laughter are dominant. Contractions relate to speaker fluency and hence are high in activation and power because the speaker is actively speaking and is not likely to yield the turn.

Backchannels are also more common in a neutral valence context than elsewhere. This again is expected because backchannels are mostly used as an acknowledgment by the listener and do not necessarily convey agreement or disagreement with the speaker. Words containing laughter are also common in the high valence contexts. One interesting case comes to light with the word *agree*, which is more common both in regions with low and with high valence than in neutral valence.

5.3 Improving Emotion Recognizers for Language Modeling

One way of improving the quality of emotion recognizer is to use more data for training. In this dissertation, I used 40 minutes of track audio that was labeled for emotion and used for training and evaluating emotion recognizers. To estimate the potential benefit of using more data, I built models that were trained using 25%, 50% and 75% of the data used for developing recognizers. The training instances were chosen randomly from the original set. To gauge the potential benefit of adding new data, each successively smaller set was a proper subset of the larger set. Figure 5.1 shows a Venn diagram of the subsets.

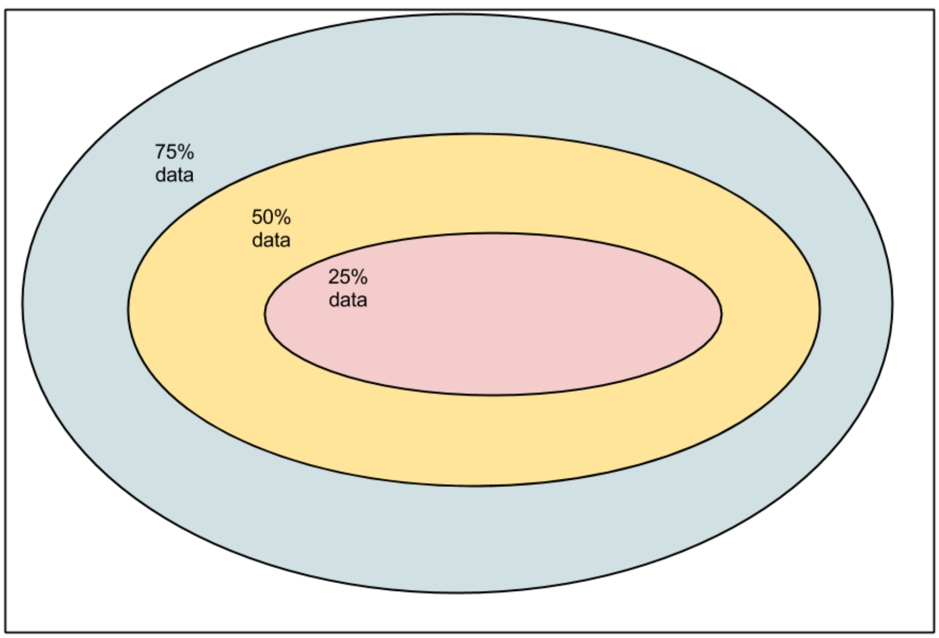

Figure 5.1: Venn diagram showing the smaller subsets chosen for emotion model testing.

The best-performing models for activation and power, mentioned in Section 3.4.3, were retrained on subsets of the original training set. Table 5.4 shows the performance of these re-trained models on activation and power as compared to the best-performing models.

Figure 5.2 shows the improvement in prediction quality, in terms of correlation, with increase in data used for training the activation recognizer. There is a linear increase in quality as the percentage of data used increases from 25% to 100%. This suggests that further improvement in recognizer's prediction is possible by using more training data.

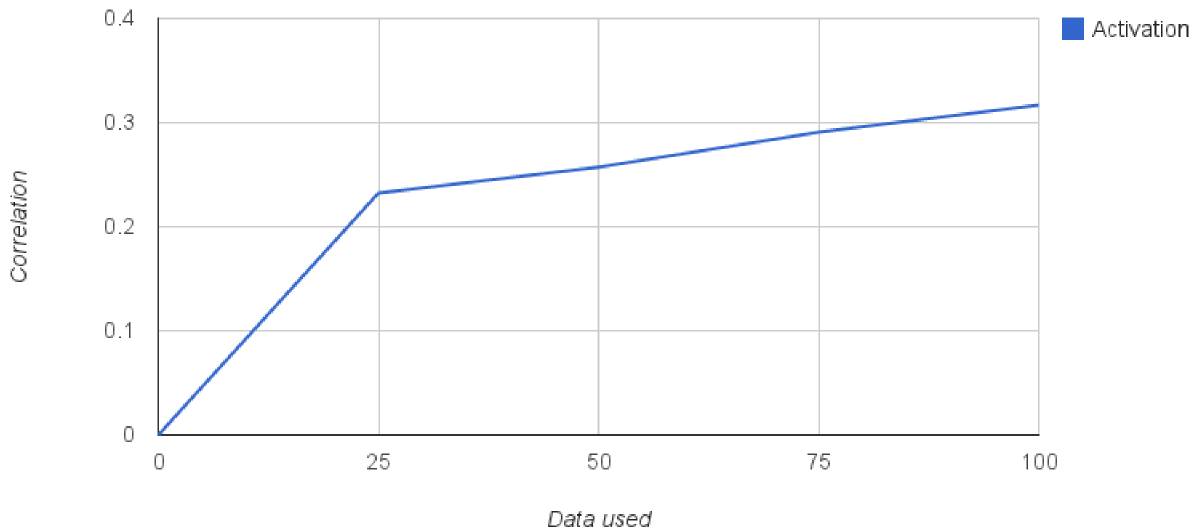

Figure 5.2: Improvement in prediction of *activation* with increase in training data.

Figure 5.3 shows the same correlation for the power recognizer. There is a linear increase in quality as the percentage of data used increases from 25% to 75%. However, the improvement seems to plateau after this point, which might indicate redundancy of the additional data used.

I left out valence from this analysis for two reasons. First, the best performing model seems to do better simply because it predicts a single value in almost all cases. This could either be due to label bias for valence or inaccurate training of the model. Second, the quality of re-trained models is proportional to the number of neutral-state (4) instances that get selected. In one of the runs, the model trained on 25% of the data out performed the rest because it predicted a value close to the neutral state label (4), which is also the

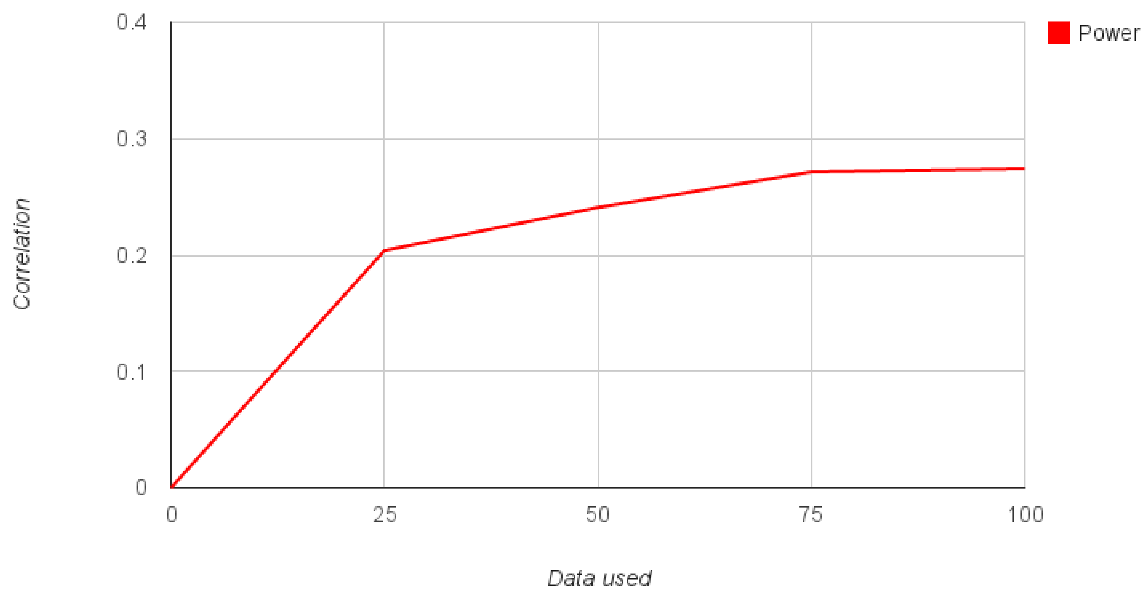

Figure 5.3: Improvement in prediction of *power* with increase in training data.

dominant label in the test set and, thus, achieves a higher correlation. By labeling more data, I expect that the effect of such a bias would be reduced if not eliminated.

5.3.1 Improvement in Language Modeling

Using the same methodology as explained in chapters 3 and 4, I developed emotion recognizers using these subsets of the original training set. The algorithms used to train the emotion recognizers were the same as the ones used for obtaining the best performing models (see Tables 3.7 and 3.9).

The credence parameter (k), was re-tuned for each dimension in isolation as described in Section 4.5.2. Table 5.5 shows the optimal-k values obtained for each dimension, in isolation, for each recognizer. Using the values specified in Table 5.5, each dimension is evaluated in isolation. Table 5.6 shows the perplexities obtained over the test set for each dimension. Figures 5.4 and 5.5 show the improvement obtained by these models.

Although, the improvement in activation recognizers showed a linear improvement with increase in data (see Figure 5.2), this is not reflected in perplexity improvement, as shown in Figure 5.4. However, there is an improvement as more data are used.

In case of power, using recognizers trained on a larger data set improved language model performance. In fact, unlike the plateauing seen in the power recognizer's prediction quality as more data is used (see Figure 5.3), there is a steeper improvement in perplexity as more data are used (as shown in Figure 5.5).

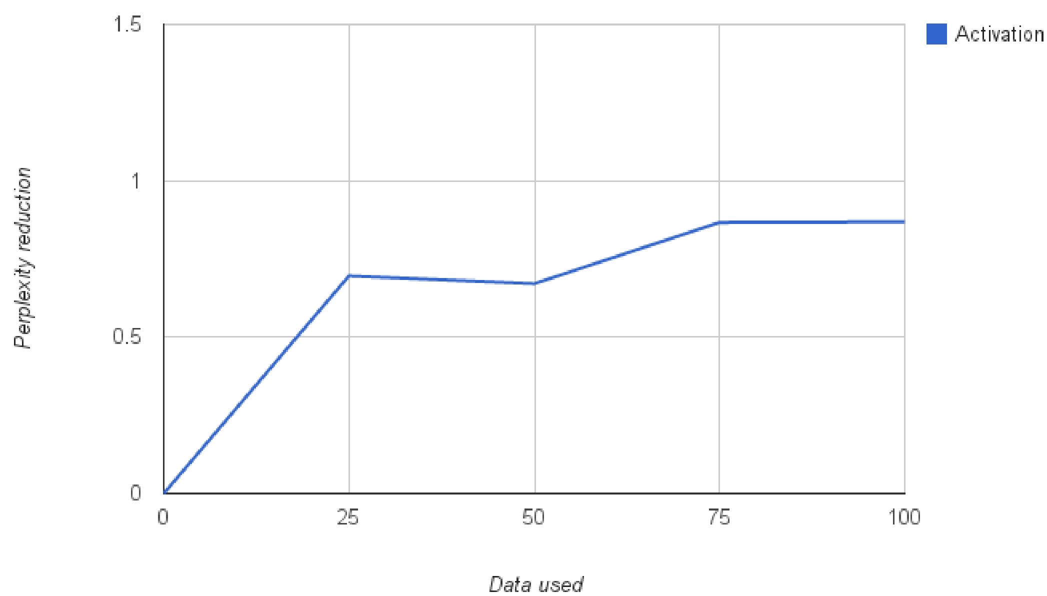

Figure 5.4: Perplexity improvement obtained using the different *activation* recognizers.

Next, the credence parameters for activation and power are tuned in combination as described in Section 4.7. Table 5.7 shows the optimal-k values for activation and power in combination. Using the values mentioned in Table 5.7, the dimensions were evaluated in combination. Table 5.8 shows the perplexity results obtained over the test set.

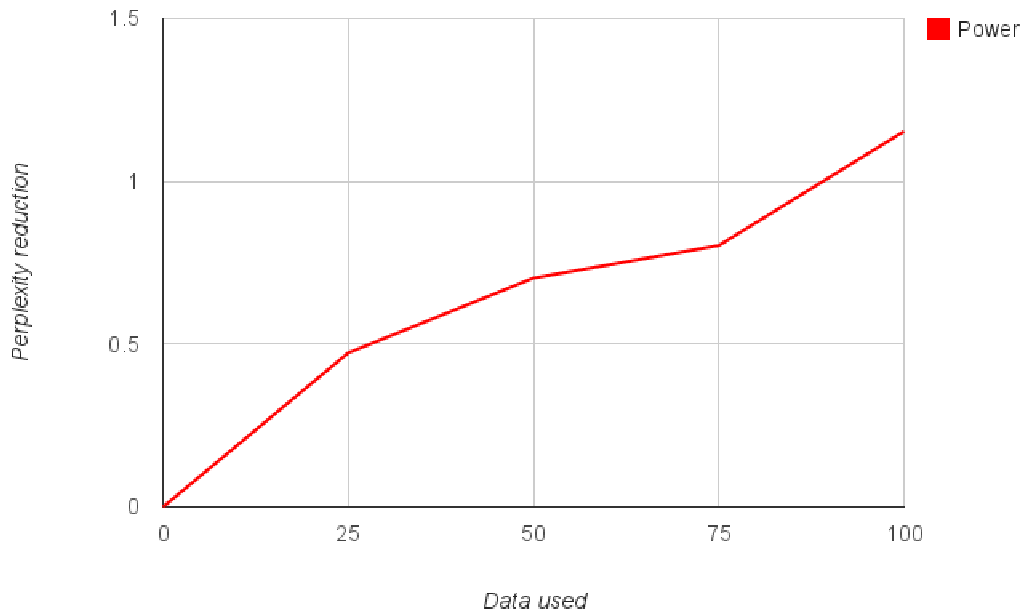

Figure 5.5: Perplexity improvement obtained using the different *power* recognizers.

Figure 5.6 shows the perplexity improvement obtained using the activation and power recognizers trained over different subsets. Using as little as 25% of the original training data, the language model shows an improvement of 0.75%. However, the rate of improvement slows down as more data are used. Activation and power recognizers trained over the original data showed an improvement of 1.14% when used in combination.

5.4 Per-word Analysis of Perplexity Improvement

To assess the effect of using an emotion recognizer on the predictive quality for each word, I computed the average perplexity change from the baseline model for each word over all of its occurrences. Tables 5.9, 5.10 and 5.11 show the top ten words, each with at least ten instances in the test set, that saw positive and negative changes to their baseline prediction from the using activation, valence and power recognizers respectively.

Table 5.9 shows that for activation the most improvement was for words that have either predominantly high or low activation (See Table 5.1).

Table 5.10 shows that backchannels and contractions see the most benefit from using the valence recognizer. With the exception of "*heard*," the words that saw a negative change had predominantly high valence values (See Table 5.2).

Table 5.11 shows that for power most improvement was observed in case of words that occur predominantly in the low power region (See Table 5.3).

In general, words that saw a positive change are more frequent than the words that saw a negative change. For example, in Table 5.9, words that saw a positive change account

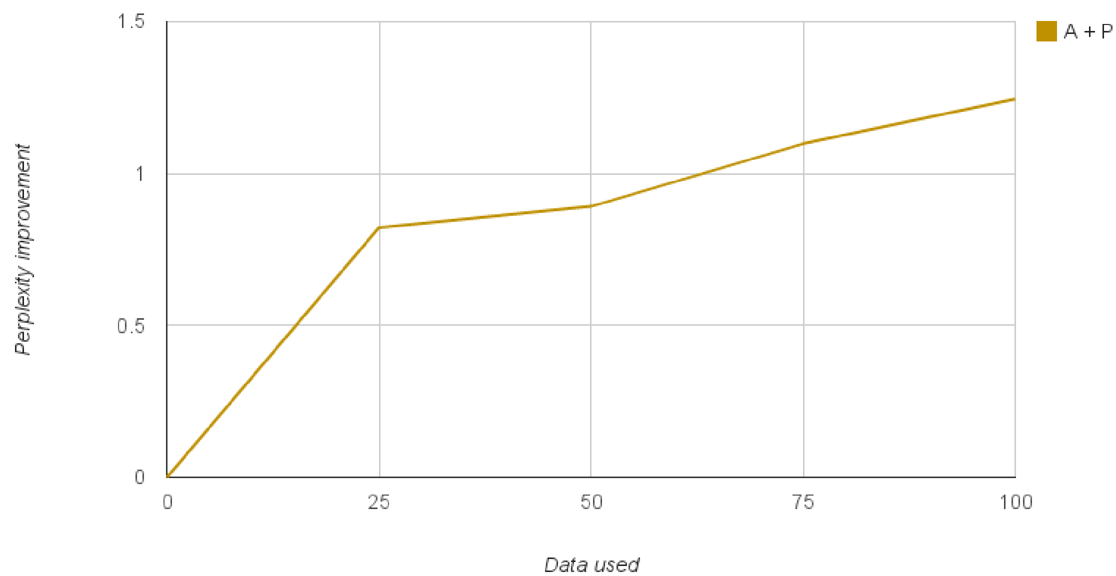

Figure 5.6: Perplexity improvement obtained using the different *activation* and *power* recognizers in combination.

for 839 instances in the test case, while words that saw a negative change account for only 221 instances.

5.5 Per-instance Analysis of Perplexity Improvement

To pinpoint instances where the emotion recognizer's predictions taken all together hurt a word's baseline estimate, I looked into changes in combined model's estimate over the n-gram for each word token in the test set. In particular, I computed the difference in log-probabilities(LP) generated by language model using emotion-related information and by the baseline language model (See Equation 5.1).

$$\Delta(LP) = log_{10}(P_{combined\ model}) - log_{10}(P_{baseline}) \tag{5.1}$$

Table 5.12 shows twenty instances where using the emotion recognizer reduced the word's n-gram estimate.

5.5.1 Causes of Anomalies

To identify speech characteristics that cause a drop in the probability estimate relative to the baseline, I listened to the instances listed in table 5.12. In particular, I listened to ten seconds of audio centered roughly around the word of interest. Most of the cases were explained by at least one of the following reasons.

1. **Atypical Usage**: Most of anomalies could be explained as atypical usage of words. For example, in case of backchannels, it is expected that a person would backchannel and continue to listen. However, in several cases, the backchannel was followed by the speaker continuing to speak (see Table 5.13 and Figure 5.7). Another example of atypical usage would be the usage of the word "*learned*." Typically it occurred in medium-activation, medium-valence and medium-power contexts in the training set. However, in this particular case the speaker seems to be passive and uninterested

in the conversation. The emotion recognizers identify the context as low-activation, medium-valence and low-power. This hurts the estimate of the token as it seldom appeared in that context in the training set.

2. **Marginal Difference between Binning Thresholds and Predicted Values**: Minor differences in binning thresholds and predicted values can cause a word token's to differ significantly. For example, tokens whose predicted values are 4.49 and 4.51 for activation would fall into different bins although their values do not differ significantly. Marginal difference between prediction and threshold values particularly affect lower-frequency words whose tokens might fall entirely in one bin.

3. **Line Disturbances**: Line disturbances in audio affects the reliability of prosodic features extracted. The unreliability in raw features in turn makes the emotion recognizer's predictions incorrect and, hence, affects the final estimate.

4. **Bleeding**: Like noise, bleeding across tracks can cause errors in prosodic feature value computation that propagate to emotion prediction inaccuracies and then to the language model, resulting in poor estimates.

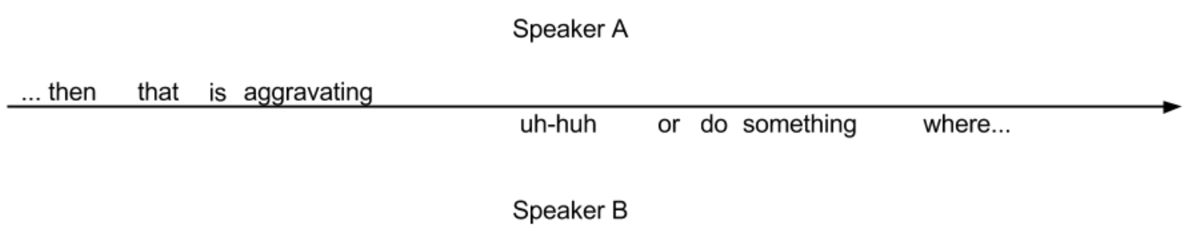

Figure 5.7: Timeline representation of exchange mentioned in Table 5.13.

5.6 Summary

Results from Chapter 4 showed that using an emotion recognizer for language modeling had only a minor improvement in language model performance. One possible reason for this might be the size of training data. In this chapter, I demonstrated that using more data for training improves the performance of emotion recognizers which, in turn, provides an improvement in perplexity.

In general, high-frequency words see a benefit on average when emotion recognizers are used. However, some high frequency utterance tokens perform worse than baseline. These instances are mostly atypical usages of words. Several other instances where the emotion-sensitive language model performs worse than baseline are attributable to bad quality audio and features that are not noise-robust. The error in prediction in these cases can be fixed by using better quality audio recordings.

Table 5.1: Top-25 words in each bin for *activation*

Low	**Medium**	**High**
hum	married	[laughter-you]
yep	nursing	[laughter-know]
wow	control	seems
huh	situation	you'd
uh-huh	computer	i'm
um-hum	public	won't
okay	drive	wouldn't
hm	top	well
yeah	give	each
um	state	i-
neat	away	read
although	cat	seem
exactly	important	although
y[ou]-	budget	you've
right	especially	don't
i[t]-	friend	i've
uh	mother	am
business	three	[laughter]
an[d]-	definitely	whole
summer	minutes	often
gun	rather	i'd
[laughter]	under	i'll
type	ten	says
matter	help	found
government	anymore	side

Table 5.2: Top-25 words in each bin for *valence*

Low	**Medium**	**High**
gone	less	[laughter-you]
set	program	[laughter-yeah]
month	business	[laughter-know]
wife	hm	heard
agree	anymore	oh
goes	um	talking
working	month	bet
sometimes	nursing	[laughter]
better	problems	well
also	um-hum	agree
um-hum	years	sounds
i-	week	that's
every	summer	knew
around	morning	no
never	drug	yes
[vocalized-noise]	minutes	mean
school	budget	hope
stuff	miles	tell
home	computer	fact
who	government	love
take	gun	found
sure	insurance	am
any	benefits	i'm
could	major	i-
has	ago	don't

Table 5.3: Top-25 words in each bin for *power*

Low	**Medium**	**High**
yep	dogs	[laughter-you]
huh	especially	[laughter-know]
um	computer	seems
um-hum	help	well
hm	man	i'm
uh-huh	second	i-
hum	minutes	mind
[laughter-yeah]	difference	wouldn't
uh	wear	found
yeah	nursing	you'd
business	both	am
exactly	situation	often
anymore	enjoy	don't
government	ago	although
[laughter]	ti	guess
recently	twenty	we'd
okay	town	let's
right	eighty	y[ou]-
morning	state	mean
today	three	feel
gun	give	[laughter]
wow	drug	hope
problems	paid	seem
water	gotten	i've
expensive	eight	each

Table 5.4: Performance of recognizers using subsets of original data on Activation, Valence and Power.

% Data	Activation	Power
100	**0.32**	**0.27**
75	0.29	0.27
50	0.26	0.24
25	0.23	0.20

Table 5.5: Optimal credence values for *activation* and *power*, in isolation, for each of the models trained on smaller sets.

% Data	Activation	Power
100	0.70	0.70
75	0.65	0.75
50	0.70	0.70
25	0.75	0.65

Table 5.6: Perplexity results obtained by using the different *activation* and *power* recognizers in isolation.

% Data	Activation	Power
100	**108.580**	**108.295**
75	108.582	108.646
50	108.778	108.746
25	108.753	108.976

Table 5.7: Optimal credence values for *activation* and *power*, in combination, for each of the models trained on smaller sets.

% Data	**Activation**	**Power**
100	0.35	0.45
75	0.4	0.50
50	0.45	0.50
25	0.6	0.35

Table 5.8: Perplexity results obtained by using the *activation* and *power* recognizers trained on different subsets. Baseline perplexity is 109.449.

% Data	**Perplexity**	**Reduction**
100	**108.204**	**1.14%**
75	108.351	1.00%
50	108.557	0.81%
25	108.627	0.75%

Table 5.9: Top-ten words that saw positive and negative change to their baseline perplexity from using the *activation* recognizer.

Positive change	**Negative change**
um-hum	basically
uh-huh	talking
um	tend
children	them_1
need	sure
great	okay
huh	enjoy
you've	getting
i'm	will
well	rather

Table 5.10: Top-ten words that saw positive and negative change to their baseline perplexity from using the *valence* recognizer.

Positive change	**Negative change**
heard	getting
um-hum	baseball
uh-huh	actually
money	basically
ago	husband
i'll	state
months	week
their	usually
working	here
night	years

Table 5.11: Top-ten words that saw positive and negative change to their baseline perplexity from using the *power* recognizer.

Positive change	**Negative change**
um-hum	basically
uh-huh	will
huh	TRUE
um	getting
well	done
enjoyed	went
children	far
ago	tend
thought	rather

Table 5.12: Top-20 instances where emotion-sensitive language model did worse than baseline. Change is shown as difference in logprob(LP) of affect-aware language model and baseline language model.

Instance Identifier	**Word**	**Δ(LP)**
sw3322A @ 44.087	um-hum	-0.831
sw2253A @ 530.231	um-hum	-0.771
sw3117B @ 398.516	hello	-0.648
sw2863B @ 136.132	uh-huh	-0.587
sw3102B @ 402.253	uh-huh	-0.587
sw3322A @ 182.773	uh-huh	-0.587
sw4718B @ 10.756	uh-huh	-0.587
sw3414B @ 228.993	um-hum	-0.581
sw2664B @ 313.534	wow	-0.539
sw3322A @ 138.078	um-hum	-0.520
sw2664B @ 393.463	wow	-0.515
sw3322A @ 204.928	um-hum	-0.491
sw3650B @ 97.428	um-hum	-0.491
sw3478B @ 246.371	anymore	-0.489
sw2253A @ 427.150	mexico	-0.488
sw3661B @ 157.677	wow	-0.478
sw2966A @ 382.568	um	-0.478
sw3414B @ 291.710	numbers	-0.461
sw3864A @ 39.183	um-hum	-0.461
sw4229A @ 259.342	learned	-0.443

Table 5.13: An Atypical use of the word "*uh-huh*".

Speaker	**Utterance**
Speaker	... then that is aggravating
Interlocutor	**uh-huh** or do something where...

Chapter 6

Conclusion and Future Work

The main claim of this dissertation was that a language model augmented with information related to the speaker's current emotional state will perform better than one which is not. The dissertation has shown that using an emotion recognizer can help in improving the performance of a language model. This work presents a novel approach of using a classless emotion recognizer trained on non-acted spontaneous speech for language modeling. Although the improvement in perplexity is small, not as strong as that provided by using raw prosody and probably not enough to improve the performance of a speech recognizer [9], there is much room for improvement in this research.

6.1 Possible Improvements

In this section, I list possible improvements to the current method of development and use of emotion recognizers in a language model.

First, as shown in Section 5.3, labeling more data is one promising approach towards developing better emotion recognizers and, in turn, better language models.

Second, I used default parameters while training emotion recognizers. It might be possible to obtain better recognizers that train over the same set of data. However, this requires detailed understanding of several machine learning algorithms. Another way of improving language model performance would be to perform feature selection prior to training recognizers.

Third, I have treated emotion as episodic – computed without reference to anything more than two seconds in the past and two seconds in the future. It is possible that a more contextualized treatment of emotion would provide a better improvement for language modeling. In particular, it would be worthwhile to explore developing models that account for speaker's previous emotional state and interlocutor's emotional state. Previous work by Acosta *et al.* [1] has shown significant correlations between speaker's and interlocutor's emotional states. In particular, the speaker's and the interlocutor's valence correlate positively, and the speaker's and the interlocutor's power correlate negatively.

Fourth, using an emotion lexicon for more information on affect might be useful in identifying words that convey key emotions. However, emotion lexicons seldom provide information for spoken language words. Another caveat with using an emotion lexicon involves handling different morphological forms of a word. Different morphologies might convey the emotion conveyed by the root form to a different degree. Handling negations might also prove challenging.

Last, experimenting with more dimensions might provide a more holistic emotion model. Smith *et al.* [40] suggest a 6-dimensional model for emotion. "Certainty" or confidence might be a dimension to add to future emotion recognizers. Other modifications could include sociolinguistic variables of interaction. For example, apical shortening of words (for example "*tryin*" instead of "*trying*") seems to indicate higher activation and power. Such variables could prove to be beneficial for future models.

6.2 Significance

I have suggested and demonstrated a unique approach towards building emotion-aware language models. Specifically, this work has shown that recognizing emotion as it occurs in normal, non-acted, spontaneous speech can improve language modeling.

6.3 Resource for Future Work

It would be ideal to deliver models of emotion recognition. However, the models have no lasting value because they are dependent on particular version of Weka and need to be re-compiled from version to version. As a tangible deliverable, I computed the mean and standard deviation for activation, valence and power for all word tokens in the language model's vocabulary over the training set. Appendix A shows the mean and standard deviations for the three dimensions for all words with at least 40 occurrences. A more comprehensive list is available at: `http://goo.gl/BB7m7`.

References

[1] J. Acosta and N. Ward. Achieving rapport with turn-by-turn, user responsive emotional coloring. *Speech Communication*, 53, 1137–1148, 2011.

[2] T. Athanaselis, S. Bakamidis, I. Dologlou, R. Cowie, E. Douglas-Cowie, and C. Cox. ASR for emotional speech: Clarifying the issues and improving performance. In *Neural Networks*, 18, 437–444, 2005.

[3] L. Bahl, P. Brown, P. de Souza and R. Mercer. A tree-based statistical language model for natural language speech recognition, *IEEE Transactions on Acoustics, Speech and Signal Processing*, 37(7):1001–1008, 1989.

[4] S. Baron-Cohen and T. Tead. *Mind Reading: The interactive guide to emotion*, Jessica Kingsley Publishers, 2003.

[5] P. Boersma. Praat, a system for doing phonetics by computer. *Glot International*, 5:9/10, 341–345.

[6] L. ten Bosch. Emotions, speech and the ASR framework. *Speech Communication*, 40, 213–225, 2003.

[7] F. Burkhart, M. van Ballegooy, R. Englert, J. Stegmann and W. Burleson. An emotion aware voice portal. In *Proc. 7th InterSpeech*, 2006.

[8] G. Caridakis, K. Karpouzis abd S. Kollias. User and context adaptive neural networks for emotion recognition. *Neurocomputing*, 71(13), 2553–2562, 2008.

[9] S. Chen, D. Beeferman and R. Rosenfeld. Evaluation metric for language models. In *DARPA Broadcast News Transcription Understanding Workshop,* 1998.

[10] S. Della Pietra, V. Della Pietra, R. Mercer and S. Roukos. Adaptive language modeling using minimum discriminant estimation, In *Proc. Speech and Natural Language DARPA Workshop*, 1992.

[11] L. DeVillers and N. Campbell. Special issue on "affective speech in real-life interactions", *Computer Speech and Language*, 25(1), 1–3, 2011.

[12] P. Ekman. Chapter: Basic Emotions. In T. Dalgleish Ed.*Handbook of cognition and emotion*, Chapter 3, 45–60, Wiley, Chichester, UK, 1999.

[13] F. Eyben, M. Wöllmer and B. Schuller. openSMILE – The Munich versatile and fast open-source audio feature extractor. In *Proceedings of ACM Multimedia*, 1459–1462, Italy, 2010.

[14] J. Fleiss, J. Cohen and B. Everitt. Large sample standard errors of kappa and weighted kappa. *Psychological Bulletin*, 72(5), 323–327, 1969.

[15] J. Godfrey, E. Holliman and J. McDaniel. SWITCHBOARD: Telephone speech corpus for research and development. In *Proc. ICASSP*, Vol. 1, pp 517–520, 1992.

[16] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. Witten. The WEKA Data Mining Software: An Update, *SIGKDD Explorations*, 11(1), 2009.

[17] 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.

[18] F. Jelinek, L. Bahl and R. Mercer. Design of a linguistic statistical decoder for the recognition of continuous speech, *IEEE Transactions on Information Theory*, 21(3):250–256, 1975.

[19] F. Jelinek. *Statistical methods for speech recognition.* MIT Press, Cambridge, MA, 1997.

[20] S. Karkhedkar and N. Ward. Representing the prosodic contexts of words using Gaussian mixture models. In *Speech Prosody 2012*, 2012.

[21] J. Kim, H. Rao, and M. Clements. Investigating the use of formant based features for detection of affective dimensions in speech, In *Affective Computing and Intelligent Interaction*, Lecture Notes in Computer Science. Springer Berlin/Heidelberg, 6975:369–377, 2011.

[22] P. Laukka, D. Neiberg, M. Forsell, I. Karlsson and K. Elenius. Expression of affect in spontaneous speech: Acoustic correlates and automatic detection of irritation and resignation. *Computer Speech and Language*, 25:84–104, 2011.

[23] R. Lazarus. Thoughts on the relations between emotion and cognition. *American Psychologist*, 37(9), 1019–1024, 1982.

[24] R. Levenson. Emotion and the autonomic nervous system: A prospectus for research on autonomic specificity. In H. L. Wagner (Ed.), *Social psychophysiology and emotion: Theory and clinical applications*, 17–42, New York, John Wiley and Sons, 1988.

[25] P. Luce and D. Pisoni. Recognizing spoken words: The neighborhood activation model. *Ear & Hearing,* 19(1), 1–36, 1998.

[26] A. Mehrabian and J. Russell. *An Approach to Environmental Psychology.* MIT Press, Cambridge, MA, 1974.

[27] H. Meng and N. Bianchi-Berthouze. Naturalistic affective expression classication by a multi-stage approach based on hidden markov models. In *Affective Computing and Intelligent Interaction*, Lecture Notes in Computer Science. Springer Berlin/Heidelberg, 6975:378–387, 2011.

[28] T. Mikolov, A. Deoras, S. Kombrink, L. Burget and J. Černocký. Empirical evaluation and combination of advanced language modeling techniques. In *Proceedings of Interspeech 2011*.

[29] G. Murray and G. Carenini. Subjectivity detection in spoken andwritten conversations. *Natural Language Engineering,* 17(3), 397–418, 2011.

[30] M. Nicolaou, H. Gunes and M. Pantic. Continuous prediction of spontaneous affect from multiple cues and modalities in valence-arousal space. *IEEE Transactions on Affective Computing,* 2(2), 92–105, 2011.

[31] D. Norris. Shortlist: a connectionist model of continuous speech recognition. *Cognition*, 52(3), 189–234, 1994.

[32] R. Plutchik. The nature of emotions. *American Scientist*, 89(4):344, 2001.

[33] J. Prinz. Which emotions are basic? In D. Evans and P. Cruse (Eds.), *Emotion, Evolution and Rationality*, Oxford University Press, 2004.

[34] R. Rosenfeld. Two decades of statistical language modeling: Where do we go from here?, *Proc. of IEEE*, 88:1270–1278, 2000.

[35] J. Russel. A circumplex of affect. *Journal of Personality and Social Psychology,* 39, 1161–1178, 1980.

[36] Y. Tausczik and J. Pennebaker. The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of Language and Social Psychology,* 29(1), 24–54, 2010.

[37] B. Schuller, J. Stadermann and G. Rigoll. Affect-robust speech recognition by dynamic emotional adaptation, In *Proc. Speech Prosody*, Special Session on Prosody in Automatic Speech Recognition, 2006.

[38] B. Schuller, M. Valstar, F. Eyben, G. McKeown, R. Cowie, and M. Pantic. Avec 2011 - the first international audio/visual emotion challenge, In *Affective Computing and Intelligent Interaction*, Lecture Notes in Computer Science. Springer Berlin/Heidelberg, 6975:415–424, 2011.

[39] T. Shikler. Analysis of affective expression in speech. PhD Thesis, Cambridge University, 2007.

[40] C. Smith and P. Ellsworth. Patterns of cognitive appraisal in emotion. *Journal of Personality and Social Psychology,* 48(4), 813–838, 1985.

[41] S. Steidl, A. Batliner, D. Seppi, and B. Schuller. On the impact of children's emotional speech on acoustic and language models. *EURASIP Journal on Audio, Speech, and Music Processing*, January 2010, 6:1–6:14.

[42] A. Stolcke, E. Shriberg, D. Hakkani-Tür and G. Tür. Modeling the prosody of hidden events for improved word recognition, In *Proc. Eurospeech*, 307–310, 1999.

[43] A. Stolcke. SRILM – An extensible language modeling toolkit. In *Proc. International Conference on Spoken Language Processing*, 2002.

[44] P. Taylor, S. King, S. Isard, H. Wright and J. Kowtko. Using intonation to constrain language models in speech recognition, In *Proc. Eurospeech*, 2763–2766, 1997.

[45] A. Vega. On the selection of prosodic features for language modeling, M.S. Thesis, University of Texas at El Paso, 2012.

[46] D. Ververidis and C. Kotropoulos. A state-of-the-art review on emotional speech databases. In *Proc. First Richmedia Conference, 109–119, Switzerland*, 2003.

[47] B. Vlasenko, D. Prylipko, and A. Wendemuth. Towards Robust Spontaneous Speech Recognition with Emotional Speech Adapted Acoustic Models. In 35^{th} *German Conference on Artificial Intelligence*, 103–107, 2012.

[48] N. Ward and R. Escalante-Ruiz. Using responsive prosodic variation to acknowledge the user's current state. In *Proceedings of Interspeech,* 2009, 2431-2434.

[49] N. Ward, A. Vega and T. Baumann. Prosodic and temporal features for language modeling for dialog. *Speech Communication*, 54:161–174, 2012.

[50] N. Ward and A. Vega. Towards empirical dialog–state modeling and its use in language modeling. In *Proceedings of Interspeech 2012*.

[51] C. Whissel. The dictionary of affect in language. In *Emotion: Theory and research*, 2005, 113–131.

[52] M. Wöllmer, F. Eyben, S. Reiter, B. Schuller, C. Cox, E. Douglas-Cowie and R. Cowie. Abandoning Emotion Classes — Towards Continuous Emotion Recognition with Modelling of Long-Range Dependencies, In *Proc. 9th Interspeech*, 597–600, 2008.

Appendix A

Affect Lexicon for Spoken Words

This appendix contains the affect lexicon for spoken words. Words with at least 40 instances in the training set are shown. For each word, I have computed the mean and standard deviation for activation, valence and power predicted for each of its tokens.

Affect lexicon

Word	Instances	Activation		Valence		Power	
		Mean	SD	Mean	SD	Mean	SD
i	22874	4.316	0.697	5.161	0.386	4.236	0.677
and	21908	4.199	0.659	5.077	0.323	4.028	0.633
the	19048	4.203	0.606	5.046	0.289	4.079	0.578
you	15999	4.201	0.693	5.104	0.344	4.108	0.669
a	14721	4.318	0.63	5.078	0.32	4.181	0.608
to	14047	4.286	0.607	5.066	0.31	4.119	0.577
uh	13505	3.946	0.686	5.009	0.241	3.795	0.68
that	13146	4.199	0.638	5.062	0.306	4.052	0.613
of	10974	4.367	0.61	5.074	0.315	4.203	0.584
it	10927	4.28	0.679	5.11	0.349	4.152	0.658
know	9435	4.166	0.694	5.078	0.322	4.098	0.689
yeah	9118	3.832	0.731	5.143	0.374	3.79	0.736
in	7966	4.233	0.603	5.055	0.296	4.067	0.578
they	6771	4.198	0.632	5.092	0.334	4.104	0.604
have	5940	4.303	0.611	5.098	0.339	4.191	0.593
but	5803	4.162	0.707	5.088	0.33	4.034	0.69
it's	5432	4.275	0.701	5.127	0.361	4.194	0.675
so	5304	4.187	0.703	5.099	0.342	4.083	0.69
we	5247	4.247	0.66	5.085	0.331	4.149	0.641
is	5161	4.236	0.635	5.081	0.323	4.096	0.62
was	4776	4.341	0.656	5.103	0.346	4.233	0.634
[laughter]	4681	4.152	0.906	5.226	0.417	3.998	0.893
like	4541	4.303	0.625	5.076	0.323	4.161	0.597
well	4478	4.31	0.797	5.219	0.416	4.297	0.809
just	4470	4.272	0.651	5.085	0.331	4.163	0.627
that's	4237	4.221	0.718	5.185	0.401	4.163	0.68
do	4065	4.233	0.667	5.112	0.352	4.12	0.644
um	4056	3.849	0.669	4.992	0.216	3.676	0.672
think	3803	4.291	0.659	5.142	0.371	4.208	0.644
oh	3754	4.048	0.748	5.23	0.425	3.981	0.73
for	3746	4.234	0.6	5.051	0.291	4.059	0.581
don't	3666	4.387	0.688	5.158	0.386	4.309	0.678
right	3402	3.888	0.693	5.116	0.361	3.794	0.666
on	3340	4.326	0.606	5.057	0.3	4.131	0.581
uh-huh	3274	3.588	0.558	5.01	0.249	3.487	0.54
or	3214	4.224	0.583	5.041	0.28	3.994	0.556
um-hum	3110	3.521	0.486	4.976	0.22	3.427	0.487
what	3005	4.273	0.668	5.114	0.351	4.176	0.64
my	2943	4.199	0.636	5.066	0.315	4.098	0.606
be	2902	4.136	0.607	5.052	0.293	4.005	0.582
not	2871	4.231	0.677	5.088	0.33	4.128	0.661
really	2841	4.162	0.657	5.077	0.325	4.067	0.624
with	2741	4.259	0.615	5.033	0.274	4.081	0.584
are	2713	4.246	0.614	5.082	0.334	4.107	0.592
there	2708	4.177	0.644	5.057	0.3	4.025	0.645
if	2585	4.306	0.673	5.132	0.364	4.203	0.649
one	2543	4.255	0.64	5.075	0.319	4.113	0.615
all	2464	4.355	0.642	5.088	0.332	4.217	0.624
i'm	2396	4.408	0.733	5.159	0.388	4.308	0.719
about	2307	4.294	0.627	5.06	0.301	4.122	0.592

get	2268	4.222	0.641	5.063	0.306	4.09	0.601
had	2264	4.264	0.593	5.079	0.322	4.145	0.572
out	2214	4.302	0.593	5.07	0.318	4.105	0.572
at	2164	4.313	0.663	5.077	0.319	4.136	0.641
he	2004	4.239	0.667	5.097	0.34	4.155	0.652
as	1964	4.242	0.651	5.054	0.307	4.088	0.611
up	1909	4.306	0.619	5.061	0.311	4.129	0.597
then	1872	4.204	0.637	5.077	0.32	4.069	0.626
this	1870	4.247	0.62	5.059	0.301	4.11	0.595
lot	1853	4.362	0.603	5.052	0.302	4.231	0.594
when	1838	4.242	0.618	5.07	0.311	4.129	0.604
go	1831	4.223	0.643	5.07	0.315	4.09	0.646
people	1809	4.195	0.61	5.043	0.283	4.08	0.603
some	1805	4.273	0.615	5.047	0.287	4.165	0.591
would	1751	4.321	0.65	5.097	0.345	4.196	0.643
good	1683	4.133	0.695	5.08	0.325	4.036	0.649
mean	1643	4.329	0.737	5.166	0.397	4.239	0.729
can	1625	4.261	0.631	5.1	0.342	4.147	0.616
because	1625	4.199	0.658	5.1	0.338	4.074	0.621
no	1603	4.133	0.762	5.185	0.398	4.09	0.733
they're	1586	4.23	0.628	5.091	0.333	4.123	0.62
got	1572	4.24	0.647	5.116	0.353	4.13	0.621
kind	1558	4.193	0.627	5.079	0.32	4.095	0.594
going	1541	4.249	0.642	5.097	0.338	4.086	0.601
now	1489	4.165	0.663	5.077	0.323	4.052	0.647
time	1419	4.201	0.602	5.039	0.278	4	0.586
i've	1353	4.368	0.685	5.162	0.386	4.262	0.674
them	1339	4.249	0.613	5.035	0.271	4.08	0.606
me	1320	4.243	0.685	5.069	0.316	4.101	0.657
too	1304	4.168	0.69	5.071	0.315	4.003	0.672
were	1287	4.3	0.629	5.067	0.321	4.15	0.605
from	1284	4.246	0.605	5.068	0.307	4.096	0.584
see	1283	4.253	0.661	5.128	0.364	4.168	0.664
been	1265	4.234	0.62	5.064	0.307	4.121	0.594
things	1261	4.257	0.6	5.023	0.259	4.091	0.585
more	1249	4.171	0.596	5.023	0.258	4.031	0.576
how	1229	4.329	0.658	5.153	0.377	4.256	0.647
where	1220	4.178	0.58	5.066	0.308	4.026	0.56
your	1214	4.308	0.612	5.077	0.333	4.177	0.586
[vocalized-noise	1204	4.071	0.73	5.083	0.339	3.964	0.712
much	1193	4.273	0.622	5.043	0.287	4.126	0.596
okay	1163	3.927	0.806	5.14	0.37	3.898	0.771
something	1146	4.216	0.629	5.056	0.298	4.063	0.603
there's	1143	4.237	0.671	5.093	0.339	4.165	0.647
she	1129	4.279	0.652	5.077	0.319	4.195	0.63
little	1102	4.357	0.632	5.043	0.287	4.201	0.611
thing	1085	4.216	0.661	5.062	0.303	4.086	0.624
here	1071	4.156	0.633	5.042	0.283	3.968	0.616
their	1057	4.233	0.606	5.022	0.255	4.068	0.564
guess	1038	4.27	0.714	5.124	0.36	4.206	0.723
very	1029	4.211	0.616	5.043	0.284	4.082	0.599
our	1017	4.283	0.577	5.077	0.319	4.126	0.548

an	1011	4.3	0.586	5.083	0.325	4.167	0.579
other	984	4.257	0.586	5.055	0.296	4.075	0.575
did	983	4.271	0.676	5.146	0.377	4.203	0.662
yes	929	3.955	0.72	5.183	0.408	3.9	0.725
i-	894	4.345	0.838	5.17	0.404	4.271	0.812
two	894	4.171	0.573	5.035	0.268	4.005	0.557
you're	878	4.304	0.676	5.131	0.366	4.215	0.682
years	863	4.146	0.596	4.982	0.184	3.968	0.576
say	823	4.275	0.661	5.113	0.354	4.16	0.646
didn't	822	4.263	0.678	5.127	0.359	4.18	0.663
work	813	4.243	0.615	5.036	0.276	4.034	0.599
we're	809	4.263	0.658	5.1	0.335	4.166	0.646
them_1	805	4.318	0.596	5.049	0.29	4.121	0.587
has	805	4.24	0.614	5.041	0.302	4.124	0.569
back	795	4.246	0.599	5.035	0.273	4.057	0.561
pretty	793	4.139	0.656	5.056	0.3	4.083	0.623
way	791	4.247	0.628	5.038	0.274	4.062	0.596
real	791	4.233	0.637	5.054	0.294	4.125	0.581
could	782	4.267	0.656	5.107	0.345	4.147	0.631
even	774	4.261	0.663	5.101	0.34	4.159	0.632
probably	768	4.126	0.628	5.094	0.334	4.038	0.596
any	763	4.225	0.63	5.065	0.303	4.059	0.605
those	761	4.222	0.62	5.06	0.301	4.088	0.575
down	748	4.169	0.572	5.033	0.271	3.997	0.547
sure	741	4.131	0.696	5.119	0.37	4.05	0.669
take	723	4.252	0.681	5.083	0.33	4.102	0.635
want	720	4.275	0.61	5.059	0.303	4.106	0.581
than	716	4.239	0.593	5.065	0.308	4.035	0.566
year	694	4.14	0.611	5.002	0.226	3.962	0.607
over	692	4.342	0.606	5.045	0.289	4.125	0.566
who	682	4.22	0.582	5.05	0.294	4.09	0.551
into	664	4.281	0.595	5.05	0.292	4.091	0.544
which	660	4.16	0.68	5.089	0.332	4.021	0.654
said	608	4.333	0.685	5.101	0.342	4.207	0.65
stuff	603	4.248	0.617	5.027	0.264	4.062	0.615
school	592	4.283	0.623	5.046	0.289	4.087	0.596
put	591	4.263	0.675	5.07	0.312	4.125	0.633
home	590	4.27	0.675	5.03	0.298	4.062	0.656
make	587	4.195	0.603	5.04	0.279	4.061	0.567
he's	587	4.242	0.665	5.104	0.348	4.114	0.656
can't	585	4.247	0.634	5.095	0.335	4.164	0.605
never	584	4.194	0.629	5.112	0.361	4.094	0.584
her	581	4.308	0.614	5.05	0.288	4.12	0.566
went	580	4.337	0.599	5.065	0.307	4.192	0.576
these	578	4.177	0.588	5.026	0.258	4.036	0.556
because_1	576	4.239	0.647	5.127	0.36	4.138	0.64
only	573	4.312	0.671	5.113	0.354	4.229	0.652
by	564	4.2	0.615	5.068	0.313	4.038	0.611
nice	545	4.182	0.702	5.1	0.343	4.114	0.659
around	536	4.277	0.6	5.042	0.318	4.077	0.566
doing	534	4.188	0.623	5.06	0.303	4.015	0.58
big	527	4.198	0.589	5.058	0.301	4.064	0.562

off	527	4.352	0.636	5.072	0.32	4.139	0.582
kids	522	4.201	0.619	5.058	0.3	4.03	0.595
him	516	4.261	0.618	5.051	0.296	4.065	0.616
anything	514	4.292	0.644	5.071	0.316	4.061	0.631
day	514	4.186	0.625	5.043	0.287	3.953	0.632
three	511	4.138	0.547	5.031	0.267	3.993	0.517
money	511	4.189	0.564	4.99	0.198	3.979	0.543
always	505	4.364	0.649	5.097	0.34	4.235	0.601
actually	498	4.203	0.692	5.097	0.335	4.084	0.671
we've	484	4.319	0.658	5.093	0.335	4.219	0.631
maybe	476	4.121	0.658	5.068	0.312	4.027	0.641
long	473	4.325	0.602	5.08	0.324	4.174	0.598
come	473	4.296	0.637	5.082	0.321	4.197	0.603
care	470	4.209	0.6	5.06	0.303	4.046	0.594
every	468	4.148	0.621	5.043	0.301	3.992	0.579
five	468	4.228	0.59	5.041	0.282	4.084	0.55
still	461	4.278	0.655	5.121	0.357	4.207	0.599
most	459	4.147	0.616	5.006	0.23	4.044	0.588
his	453	4.261	0.616	5.062	0.296	4.106	0.581
used	449	4.296	0.6	5.058	0.298	4.19	0.583
us	445	4.361	0.653	5.079	0.323	4.163	0.649
will	441	4.301	0.619	5.072	0.319	4.162	0.577
last	438	4.331	0.61	5.038	0.274	4.171	0.57
first	435	4.307	0.628	5.077	0.319	4.159	0.617
getting	435	4.113	0.595	5.053	0.298	3.989	0.592
should	434	4.279	0.647	5.093	0.334	4.148	0.659
everything	433	4.19	0.577	5.032	0.269	3.973	0.535
many	433	4.269	0.611	5.035	0.275	4.129	0.595
bit	433	4.234	0.579	5.031	0.272	4.031	0.561
different	428	4.135	0.594	5.051	0.294	3.967	0.555
haven't	422	4.268	0.614	5.128	0.36	4.192	0.613
feel	422	4.389	0.625	5.122	0.35	4.305	0.641
done	419	4.132	0.602	5.052	0.298	3.995	0.578
use	416	4.235	0.606	5.037	0.277	4.114	0.6
great	407	4.069	0.679	5.091	0.336	4.017	0.648
through	399	4.231	0.607	5.048	0.286	4.041	0.571
thought	399	4.34	0.658	5.158	0.385	4.234	0.645
also	398	4.234	0.637	5.041	0.281	4.085	0.597
old	395	4.299	0.613	5.076	0.32	4.101	0.61
children	394	4.131	0.604	5.031	0.269	3.986	0.606
course	391	4.225	0.706	5.071	0.316	4.14	0.658
problem	390	4.231	0.672	5.088	0.329	4.105	0.654
sort	383	4.204	0.63	5.061	0.297	4.085	0.602
before	379	4.176	0.62	5.047	0.284	3.985	0.596
same	372	4.25	0.611	5.056	0.301	4.092	0.579
pay	371	4.155	0.636	5.042	0.282	3.965	0.568
family	369	4.16	0.576	5.056	0.296	4	0.559
being	368	4.132	0.582	5.043	0.279	3.984	0.567
does	368	4.186	0.643	5.078	0.318	4.131	0.614
huh	366	3.728	0.704	5.053	0.296	3.609	0.653
trying	366	4.227	0.618	5.065	0.308	4.142	0.552
need	365	4.172	0.536	5.061	0.3	4.031	0.525

i'd	357	4.302	0.707	5.087	0.327	4.2	0.691
house	357	4.313	0.568	5.06	0.3	4.109	0.563
whole	357	4.42	0.645	5.088	0.333	4.288	0.64
car	352	4.18	0.626	5.048	0.29	3.962	0.596
four	348	4.196	0.587	5.064	0.306	4.018	0.564
keep	346	4.253	0.678	5.105	0.351	4.167	0.638
hard	346	4.289	0.628	5.083	0.322	4.189	0.582
quite	345	4.095	0.632	5.023	0.255	3.971	0.574
interesting	343	4.112	0.688	5.096	0.337	4.089	0.658
talking	342	4.332	0.582	5.242	0.424	4.211	0.552
ever	341	4.282	0.597	5.127	0.362	4.13	0.565
doesn't	340	4.17	0.647	5.072	0.319	4.062	0.61
though	340	4.225	0.712	5.081	0.328	4.071	0.719
bad	337	4.172	0.679	5.067	0.311	4.028	0.614
seems	337	4.43	0.72	5.121	0.358	4.361	0.732
why	334	4.288	0.719	5.091	0.333	4.19	0.685
area	332	4.058	0.625	5.016	0.249	3.855	0.613
live	331	4.31	0.645	5.051	0.292	4.204	0.606
i'll	329	4.362	0.632	5.141	0.371	4.236	0.607
look	328	4.32	0.615	5.049	0.294	4.205	0.6
hundred	325	4.255	0.625	5.052	0.291	4.103	0.583
another	320	4.253	0.604	5.095	0.334	4.081	0.589
place	319	4.203	0.603	5.027	0.268	4.016	0.578
try	311	4.179	0.592	5.056	0.295	4.046	0.569
fact	310	4.303	0.676	5.143	0.372	4.167	0.647
twenty	306	4.159	0.557	5.057	0.294	3.993	0.522
they've	305	4.242	0.699	5.088	0.332	4.144	0.642
seen	303	4.271	0.555	5.114	0.35	4.16	0.506
enough	303	4.28	0.628	5.058	0.299	4.129	0.604
she's	303	4.244	0.599	5.048	0.293	4.133	0.598
far	301	4.28	0.61	5.033	0.269	4.188	0.603
either	293	4.13	0.608	5.089	0.331	3.98	0.6
find	291	4.348	0.677	5.155	0.381	4.237	0.65
having	291	4.197	0.578	5.055	0.296	4.061	0.564
part	290	4.179	0.635	5.057	0.3	4.035	0.61
dollars	289	4.233	0.58	5.037	0.275	4.022	0.564
away	286	4.29	0.604	5.044	0.287	4.061	0.611
few	286	4.219	0.608	5.031	0.263	4.081	0.581
made	285	4.243	0.605	5.093	0.33	4.083	0.579
whatever	283	4.132	0.569	5.021	0.258	3.888	0.577
tell	282	4.301	0.657	5.177	0.396	4.21	0.652
better	281	4.165	0.688	5.086	0.329	3.994	0.647
usually	279	4.201	0.614	5.061	0.301	4.09	0.613
else	278	4.243	0.613	5.05	0.293	4.029	0.606
while	276	4.262	0.736	5.048	0.29	4.071	0.711
after	274	4.243	0.634	5.073	0.315	4.093	0.577
new	271	4.245	0.638	5.011	0.239	4.078	0.576
watch	270	4.272	0.587	5.057	0.298	4.147	0.567
couple	270	4.191	0.565	5.046	0.285	4.053	0.584
ago	270	4.266	0.525	4.998	0.223	4.044	0.486
once	266	4.196	0.593	5.04	0.285	4.028	0.575
give	264	4.088	0.512	5.053	0.297	3.961	0.496

wasn't	264	4.289	0.664	5.086	0.325	4.199	0.604
sometimes	261	4.274	0.651	5.071	0.36	4.147	0.617
high	261	4.201	0.648	5.016	0.245	4.062	0.598
six	260	4.17	0.594	5.046	0.283	4.032	0.582
job	260	4.157	0.647	5.01	0.241	3.981	0.608
gonna	258	4.262	0.605	5.087	0.329	4.117	0.575
times	257	4.219	0.576	5.04	0.281	4.04	0.539
least	256	4.313	0.669	5.054	0.298	4.195	0.658
heard	255	4.391	0.625	5.202	0.404	4.278	0.606
country	252	4.048	0.541	5.037	0.275	3.834	0.55
start	252	4.234	0.657	5.052	0.286	4.135	0.565
own	252	4.395	0.545	5.092	0.33	4.14	0.524
somebody	251	4.195	0.601	5.06	0.3	4.089	0.532
ones	250	4.293	0.681	5.041	0.279	4.128	0.669
what's	248	4.28	0.713	5.146	0.381	4.163	0.685
type	247	4.134	0.68	5.029	0.261	3.986	0.644
wouldn't	246	4.389	0.672	5.164	0.386	4.285	0.681
week	245	4.186	0.61	4.998	0.227	3.942	0.592
might	245	4.173	0.637	5.066	0.311	4.061	0.593
call	243	4.261	0.573	5.062	0.305	4.12	0.603
again	242	4.195	0.617	5.073	0.318	3.999	0.632
life	238	4.317	0.567	5.031	0.264	4.071	0.579
remember	237	4.321	0.571	5.164	0.382	4.22	0.576
anyway	237	4.153	0.752	5.06	0.301	3.979	0.742
started	235	4.314	0.616	5.086	0.333	4.18	0.561
talk	234	4.295	0.697	5.123	0.358	4.161	0.667
buy	234	4.112	0.637	5.068	0.309	3.973	0.645
ten	229	4.108	0.556	5.044	0.284	3.921	0.523
love	228	4.412	0.637	5.157	0.376	4.328	0.603
am	228	4.432	0.699	5.186	0.401	4.292	0.704
exactly	227	3.839	0.654	5.135	0.365	3.744	0.664
able	226	4.299	0.596	5.019	0.254	4.139	0.58
college	224	4.178	0.62	5.003	0.224	3.934	0.611
let	224	4.32	0.638	5.089	0.331	4.189	0.609
working	223	4.197	0.648	5.025	0.338	4.019	0.637
husband	223	4.307	0.611	5.096	0.33	4.208	0.554
person	223	4.136	0.597	4.99	0.2	4.002	0.552
end	222	4.266	0.614	5.082	0.323	4.111	0.583
came	222	4.358	0.673	5.109	0.34	4.233	0.629
fun	222	4.221	0.673	5.114	0.352	4.157	0.603
almost	220	4.253	0.645	5.081	0.327	4.117	0.567
you've	217	4.455	0.61	5.13	0.36	4.333	0.576
saw	212	4.299	0.617	5.148	0.379	4.244	0.606
read	212	4.344	0.681	5.091	0.337	4.149	0.651
believe	212	4.257	0.585	5.086	0.328	4.158	0.561
since	211	4.17	0.564	5.046	0.285	4.054	0.599
point	211	4.168	0.628	5.052	0.288	4.03	0.597
someone	210	4.161	0.572	5.029	0.271	4.047	0.558
may	203	4.216	0.608	5.076	0.317	4.136	0.604
hm	203	3.569	0.546	4.971	0.169	3.483	0.533
problems	202	4.152	0.626	5.002	0.228	3.956	0.643
gets	202	4.242	0.604	5.072	0.314	4.127	0.599

everybody	201	4.2	0.682	5.118	0.352	4.08	0.653
they'll	201	4.256	0.659	5.123	0.361	4.144	0.609
parents	200	4.287	0.65	5.091	0.333	4.111	0.638
isn't	199	4.303	0.654	5.157	0.389	4.208	0.621
both	199	4.181	0.583	5.043	0.285	4.066	0.541
movie	196	4.142	0.68	5.012	0.243	4.019	0.62
next	195	4.329	0.607	5.071	0.312	4.115	0.581
system	195	4.17	0.579	4.999	0.217	3.976	0.572
thousand	194	4.27	0.583	5.013	0.243	4.125	0.567
enjoy	193	4.242	0.563	5.079	0.321	4.1	0.552
yet	193	4.134	0.705	5.042	0.285	3.943	0.678
until	193	4.248	0.661	5.087	0.331	4.115	0.634
idea	193	4.222	0.567	5.074	0.318	4.08	0.584
took	192	4.286	0.602	5.054	0.288	4.161	0.622
goes	190	4.22	0.637	5.018	0.328	4.05	0.584
play	190	4.183	0.609	5.035	0.269	4.003	0.589
agree	188	4.282	0.673	5.184	0.462	4.172	0.691
looking	187	4.263	0.695	5.046	0.291	4.11	0.677
couldn't	187	4.324	0.713	5.152	0.38	4.216	0.652
wanted	186	4.263	0.572	5.055	0.295	4.1	0.54
called	186	4.275	0.62	5.089	0.332	4.142	0.577
night	185	4.152	0.638	5.015	0.249	3.935	0.606
run	185	4.267	0.649	5.036	0.283	4.078	0.614
food	185	4.286	0.682	5.043	0.286	4.046	0.659
half	182	4.211	0.674	5.077	0.32	4.022	0.65
makes	182	4.206	0.66	5.135	0.362	4.115	0.625
state	182	4.166	0.53	5.013	0.23	4.011	0.51
saying	181	4.238	0.637	5.054	0.299	4.104	0.614
company	181	4.099	0.586	5.017	0.249	3.891	0.598
days	179	4.054	0.567	4.988	0.186	3.931	0.524
let's	179	4.284	0.759	5.106	0.341	4.204	0.77
spend	179	4.25	0.566	5.069	0.309	4.114	0.558
child	177	4.228	0.642	5.05	0.283	4.001	0.643
each	174	4.436	0.619	5.096	0.332	4.257	0.641
such	173	4.301	0.562	5.033	0.271	4.124	0.531
dog	172	4.151	0.656	5.031	0.27	3.96	0.61
news	171	4.182	0.65	5.025	0.263	3.943	0.586
water	170	4.218	0.679	5.035	0.28	4.006	0.644
myself	169	4.242	0.661	5.047	0.293	4.047	0.658
gone	169	4.211	0.609	5.044	0.448	4.096	0.578
month	169	4.204	0.605	4.998	0.226	3.968	0.577
sounds	168	4.255	0.657	5.177	0.389	4.298	0.649
understand	167	4.284	0.615	5.129	0.364	4.155	0.61
worked	166	4.255	0.632	5.07	0.315	4.147	0.607
guy	166	4.13	0.594	5.084	0.326	4.007	0.575
thirty	166	4.1	0.71	5.097	0.34	3.991	0.672
eight	166	4.152	0.587	5.053	0.292	3.981	0.533
best	165	4.229	0.623	5.049	0.289	4.055	0.565
wife	165	4.315	0.603	5.042	0.287	4.161	0.609
wonderful	165	4.114	0.639	5.065	0.305	4.075	0.587
supposed	165	4.306	0.525	5.072	0.318	4.24	0.493
whether	165	4.193	0.532	5.029	0.264	4.018	0.525

music	163	4.129	0.62	5.014	0.233	3.942	0.658
help	161	4.17	0.514	5.032	0.269	4.031	0.5
wow	161	3.633	0.651	5.034	0.28	3.626	0.65
paper	160	4.089	0.641	5.039	0.276	3.894	0.646
insurance	158	4.226	0.602	5.021	0.26	3.961	0.572
hear	156	4.291	0.563	5.064	0.298	4.201	0.575
comes	156	4.286	0.619	5.093	0.335	4.14	0.62
thinking	156	4.273	0.638	5.044	0.279	4.165	0.583
lived	156	4.306	0.559	5.102	0.339	4.2	0.524
matter	153	4.088	0.706	5.075	0.326	4.011	0.672
name	153	4.19	0.637	5.06	0.303	4.088	0.643
found	153	4.362	0.728	5.142	0.362	4.219	0.746
government	153	4.051	0.63	5	0.221	3.865	0.594
yep	152	3.603	0.655	5.073	0.324	3.508	0.668
basically	152	4.012	0.609	4.975	0.177	3.899	0.584
coming	152	4.278	0.608	5.105	0.344	4.123	0.608
taking	151	4.139	0.582	5.056	0.299	4.001	0.57
small	150	4.186	0.57	5.044	0.281	4.042	0.523
bought	150	4.214	0.563	5.054	0.293	4.068	0.551
places	149	4.181	0.606	4.99	0.211	4.026	0.58
[laughter-yeah]	148	4.023	0.792	5.324	0.445	3.924	0.822
i[t]-	148	4.021	0.727	5.121	0.359	3.984	0.682
neat	147	3.968	0.784	5.047	0.284	3.986	0.716
health	147	4.276	0.661	4.991	0.207	4.04	0.577
stay	147	4.226	0.702	5.027	0.253	4.098	0.649
together	146	4.167	0.619	5.01	0.241	3.949	0.623
boy	146	4.07	0.654	5.044	0.282	4.002	0.612
credit	145	4.373	0.645	5.063	0.307	4.15	0.523
tried	145	4.239	0.642	5.115	0.357	4.182	0.581
months	144	4.223	0.595	4.995	0.218	4.023	0.593
schools	142	4.216	0.562	5.025	0.266	4.042	0.538
especially	142	4.074	0.564	5.078	0.328	3.95	0.537
during	142	4.108	0.575	5.03	0.273	3.945	0.579
seem	141	4.263	0.675	5.147	0.369	4.233	0.607
number	141	4.218	0.686	5.034	0.273	4.037	0.627
set	141	4.219	0.63	5.028	0.261	4.105	0.563
percent	141	4.26	0.622	5.046	0.285	4.057	0.594
crime	140	4.111	0.664	5.05	0.279	3.939	0.637
world	140	4.256	0.639	5.006	0.24	4.088	0.605
we'll	138	4.26	0.598	5.103	0.347	4.127	0.564
definitely	138	3.997	0.546	5.095	0.341	3.842	0.521
happen	137	4.201	0.68	5.073	0.311	4.043	0.64
friends	137	4.349	0.618	5.107	0.341	4.18	0.575
gotten	135	4.132	0.567	5.062	0.308	4.005	0.527
certain	135	4.192	0.66	5.045	0.284	4.045	0.594
forty	134	4.195	0.561	5.046	0.289	4.025	0.565
jury	134	4.173	0.651	5.038	0.271	3.946	0.624
program	134	4.091	0.553	4.965	0.14	3.88	0.552
tax	133	4.286	0.588	5.014	0.249	4.088	0.566
taxes	133	4.213	0.629	5.032	0.275	4.036	0.589
expensive	132	4.231	0.657	5.086	0.326	4.035	0.659
easy	131	4.298	0.689	5.119	0.364	4.159	0.673

certainly	131	4.335	0.705	5.116	0.35	4.206	0.667
show	130	4.216	0.577	5.018	0.259	4.089	0.587
bye-bye	130	3.754	1.062	5.265	0.425	3.621	0.836
eat	130	4.331	0.676	5.069	0.318	4.152	0.661
important	129	4.132	0.557	5	0.22	3.961	0.554
you'd	129	4.397	0.659	5.133	0.364	4.272	0.659
business	129	4.058	0.671	4.987	0.202	3.875	0.64
change	128	4.211	0.627	5.051	0.292	4.031	0.56
paying	128	4.314	0.633	5.07	0.305	4.077	0.578
won't	128	4.416	0.691	5.098	0.34	4.269	0.631
close	127	4.171	0.675	5.05	0.288	4.054	0.621
funny	127	4.108	0.619	5.118	0.356	4.047	0.628
aren't	126	4.336	0.626	5.093	0.334	4.152	0.629
cars	126	4.106	0.626	4.992	0.209	3.908	0.616
public	126	4.112	0.539	5.011	0.239	3.905	0.505
hum	125	3.524	0.568	5.006	0.225	3.464	0.535
several	125	4.218	0.62	5.018	0.254	4.107	0.593
gun	125	4.022	0.562	4.992	0.205	3.904	0.581
moved	125	4.176	0.554	5.023	0.252	4.043	0.534
reason	125	4.31	0.53	5.033	0.277	4.151	0.552
young	125	4.173	0.63	5.014	0.245	3.981	0.533
deal	124	4.161	0.591	5.042	0.287	4.105	0.616
today	124	4.134	0.704	5.08	0.33	3.948	0.686
camping	124	4.193	0.593	5.02	0.255	3.972	0.578
y[ou]-	124	4.224	0.801	5.135	0.366	4.147	0.83
town	124	4.168	0.565	5.023	0.251	3.952	0.55
women	124	4.203	0.545	5.021	0.26	3.97	0.539
rather	123	4.12	0.539	5.08	0.327	3.962	0.516
movies	122	4.243	0.589	5.08	0.32	4.053	0.613
side	122	4.312	0.594	5	0.229	4.112	0.652
mine	122	4.27	0.653	5.13	0.365	4.143	0.651
wear	122	4.258	0.541	5.024	0.264	4.089	0.5
nothing	122	4.177	0.631	5.056	0.306	4.03	0.664
anymore	122	4.089	0.572	4.984	0.191	3.806	0.569
eighty	122	4.082	0.537	5.028	0.268	3.914	0.52
along	121	4.315	0.579	5.063	0.311	4.197	0.53
nursing	121	4.017	0.545	4.98	0.187	3.829	0.442
hand	120	4.332	0.658	5.075	0.313	4.097	0.606
paid	119	4.256	0.564	5.087	0.321	4.127	0.502
fifty	119	4.278	0.549	5.072	0.319	4.141	0.488
summer	119	3.926	0.587	4.999	0.224	3.746	0.508
making	118	4.196	0.555	5.043	0.281	4.045	0.531
situation	118	4.052	0.558	5.029	0.266	3.911	0.544
older	118	4.27	0.602	5.023	0.256	4.111	0.567
education	118	4.178	0.613	5.007	0.229	3.95	0.568
seven	117	4.2	0.572	5.049	0.287	4.068	0.582
case	117	4.187	0.631	5.008	0.224	4.04	0.632
involved	117	4.101	0.57	4.989	0.201	3.944	0.558
control	117	4.183	0.533	5.03	0.261	4.038	0.561
miles	115	4.231	0.599	4.98	0.182	4.038	0.611
married	115	4.127	0.489	5.01	0.238	3.989	0.495
cat	114	4.124	0.56	4.974	0.16	3.915	0.501

[laughter-know]	114	4.466	0.958	5.311	0.442	4.321	0.967
computer	114	4.147	0.587	5.024	0.266	3.905	0.536
mother	113	4.207	0.587	5.038	0.281	4.076	0.577
man	113	4.234	0.585	5.069	0.314	4.104	0.543
living	113	4.201	0.621	4.986	0.191	4.059	0.57
turn	113	4.229	0.61	5.07	0.308	4.077	0.525
friend	113	4.236	0.516	5.187	0.401	4.113	0.515
knew	113	4.326	0.674	5.142	0.366	4.17	0.683
drive	113	4.131	0.56	5.045	0.293	3.941	0.557
although	113	4.231	0.838	5.051	0.299	4.155	0.775
kinds	113	4.233	0.552	5.046	0.281	4.036	0.522
daughter	112	4.177	0.595	5.082	0.326	4.013	0.589
takes	112	4.187	0.666	5.078	0.325	4.085	0.638
must	111	4.338	0.57	5.139	0.374	4.28	0.594
twelve	111	4.197	0.632	5.049	0.29	4.025	0.655
often	111	4.393	0.692	5.086	0.334	4.149	0.664
fifteen	111	4.152	0.673	4.996	0.208	4	0.633
works	111	4.306	0.629	5.046	0.291	4.114	0.666
talked	110	4.33	0.619	5.144	0.373	4.223	0.548
says	110	4.373	0.639	5.072	0.303	4.263	0.639
gosh	110	3.992	0.744	5.095	0.338	3.918	0.663
minutes	110	4.201	0.502	5.027	0.266	4.008	0.501
under	109	4.26	0.709	5.095	0.337	4.081	0.675
room	109	4.261	0.627	5.061	0.305	4.013	0.604
top	109	4.172	0.538	5.022	0.264	4.013	0.512
city	109	4.121	0.607	4.992	0.198	3.917	0.567
outside	109	4.176	0.572	5.037	0.273	3.978	0.529
nine	109	4.03	0.559	5.036	0.281	3.893	0.527
sit	109	4.302	0.5	5.082	0.332	4.098	0.526
less	109	4.231	0.65	4.974	0.17	4.06	0.593
we'd	107	4.365	0.683	5.109	0.347	4.239	0.644
lives	107	4.25	0.637	5.008	0.221	4.086	0.544
mind	106	4.324	0.722	5.147	0.379	4.234	0.675
spent	106	4.267	0.574	5.044	0.281	4.097	0.606
hours	106	4.312	0.545	5.023	0.256	4.088	0.545
ahead	106	4.225	0.804	5.065	0.309	4.034	0.712
pick	106	4.199	0.649	5.032	0.271	4.023	0.579
absolutely	106	4.017	0.712	5.148	0.38	3.923	0.683
son	105	4.293	0.574	5.04	0.272	4.138	0.523
benefits	105	4.099	0.688	4.996	0.214	3.928	0.635
age	105	4.217	0.59	5.063	0.314	3.996	0.602
second	104	4.278	0.589	5.018	0.256	4.088	0.508
recently	104	4.023	0.648	4.974	0.174	3.862	0.672
an[d]-	104	4.036	0.604	5.093	0.335	3.913	0.605
recycling	103	4.162	0.652	5.006	0.225	3.964	0.629
bet	103	4.056	0.683	5.187	0.395	4.044	0.666
hope	103	4.409	0.638	5.192	0.402	4.313	0.647
watching	103	4.201	0.671	5.026	0.264	4.026	0.64
difference	102	4.086	0.561	5.017	0.245	3.914	0.544
major	102	4.16	0.581	5.028	0.273	3.989	0.563
without	102	4.225	0.684	5.042	0.289	4.085	0.617
line	102	4.233	0.535	5.06	0.3	4.034	0.47

[laughter-you]	102	4.716	1.039	5.333	0.448	4.551	1
dogs	101	4.144	0.547	5.08	0.329	3.958	0.502
drug	101	4.17	0.631	4.999	0.222	3.988	0.576
listen	100	4.333	0.638	5.04	0.282	4.155	0.594
interest	100	4.19	0.591	5.013	0.245	4.008	0.559
between	100	4.04	0.523	4.994	0.209	3.888	0.525
budget	100	4.077	0.571	4.988	0.195	3.869	0.522
weeks	99	4.177	0.678	4.994	0.212	3.993	0.674
lots	99	4.255	0.572	5.067	0.311	4.129	0.602
card	99	4.145	0.63	5.052	0.289	3.904	0.627
morning	99	4.097	0.648	5.038	0.279	3.865	0.645
fish	99	4.27	0.679	5.051	0.29	4.076	0.635
anybody	99	4.27	0.679	5.081	0.324	4.063	0.637
[laughter-i]	98	4.902	0.991	5.409	0.461	4.766	0.917
middle	98	4.312	0.589	5.031	0.265	4.178	0.588
favorite	98	4.246	0.607	5.098	0.332	4.155	0.57
guys	97	4.257	0.502	5.106	0.328	4.145	0.516
sense	97	4.338	0.734	5.085	0.33	4.182	0.721
wanna	97	4.397	0.646	5.036	0.269	4.214	0.666
question	97	4.357	0.734	5.106	0.343	4.214	0.708
bring	97	4.187	0.6	5.005	0.227	4.04	0.568
told	96	4.379	0.591	5.142	0.372	4.244	0.621
hour	96	4.33	0.536	5.077	0.32	4.016	0.554
local	96	4.159	0.676	5.002	0.233	3.991	0.631
drugs	95	4.134	0.635	4.98	0.186	3.912	0.614
happened	95	4.335	0.635	5.144	0.37	4.147	0.682
leave	94	4.237	0.642	5.024	0.263	4.071	0.601
instead	94	4.212	0.64	5.077	0.311	4.05	0.575
wrong	93	4.367	0.623	5.004	0.231	4.195	0.612
service	93	4.246	0.662	5.016	0.258	4.017	0.611
enjoyed	93	4.482	0.695	5.263	0.435	4.36	0.664
growing	93	4.234	0.696	5.083	0.321	4.161	0.678
cats	93	3.994	0.554	5.061	0.294	3.884	0.55
yourself	92	4.118	0.673	5.015	0.241	3.908	0.603
early	92	4.015	0.581	5.055	0.287	3.86	0.632
amount	92	4.201	0.59	5	0.231	4.041	0.506
changed	91	4.261	0.57	5.078	0.309	4.096	0.542
against	91	4.262	0.552	4.973	0.175	4.021	0.503
already	90	4.274	0.582	5.066	0.308	4.088	0.544
left	90	4.301	0.643	5.049	0.292	4.115	0.61
war	90	4.235	0.637	4.985	0.189	3.941	0.673
wait	90	4.356	0.7	5.169	0.387	4.174	0.672
mostly	90	4.021	0.653	4.989	0.197	3.906	0.625
large	90	4.258	0.666	5.041	0.285	4.065	0.674
they'd	90	4.242	0.69	5.072	0.306	4.11	0.615
cold	89	4.164	0.613	5.075	0.328	4.021	0.564
past	89	4.142	0.537	5.026	0.251	3.978	0.552
game	89	4.099	0.608	5.046	0.289	3.97	0.658
walk	89	4.314	0.671	5.046	0.291	4.144	0.644
fairly	88	4.183	0.621	5.029	0.256	4.133	0.566
putting	88	4.065	0.636	5.05	0.29	3.98	0.616
full	87	4.249	0.605	5.03	0.274	4.135	0.579

experience	87	4.225	0.635	5.04	0.271	3.996	0.525
taken	87	4.199	0.588	5.049	0.278	4.067	0.554
cost	86	4.088	0.6	5.01	0.247	3.968	0.541
check	86	4.229	0.626	5.025	0.269	4.12	0.604
happens	86	4.147	0.6	5.134	0.369	4.053	0.549
cut	85	4.3	0.631	5.106	0.341	4.115	0.64
imagine	85	4.212	0.734	5.148	0.379	4.154	0.706
liked	85	4.288	0.667	5.13	0.347	4.19	0.614
companies	84	4.118	0.638	4.985	0.194	3.932	0.574
front	84	4.255	0.518	5.039	0.283	4.061	0.514
areas	84	4.204	0.585	4.98	0.192	4.018	0.558
gave	83	4.053	0.544	5.005	0.236	3.961	0.51
topic	83	4.201	0.751	5.052	0.286	4.027	0.727
looked	83	4.464	0.556	5.105	0.344	4.36	0.599
wonder	82	4.192	0.675	5.134	0.37	4.171	0.639
looks	82	4.348	0.618	5.088	0.329	4.262	0.674
difficult	82	4.115	0.639	5.086	0.341	4.078	0.616
weather	82	4.042	0.53	4.975	0.175	3.893	0.485
office	81	4.31	0.622	5.014	0.254	4.106	0.589
free	81	4.261	0.504	5.017	0.256	4.035	0.489
rest	81	4.268	0.644	5.06	0.299	4.114	0.593
ask	81	4.514	0.661	5.15	0.376	4.333	0.6
ready	81	4.444	0.672	5.12	0.356	4.258	0.642
hot	80	4.381	0.698	5.06	0.283	4.255	0.619
air	80	4.036	0.612	5.039	0.272	3.864	0.624
death	80	4.215	0.687	5.011	0.247	4.021	0.64
grew	80	4.244	0.573	5.105	0.335	4.255	0.614
tend	80	4.098	0.603	5.058	0.294	4.004	0.575
lately	80	4.137	0.624	4.997	0.209	3.922	0.628
radio	79	4.2	0.586	5.055	0.301	3.939	0.618
decided	79	4.152	0.574	5.018	0.258	4.045	0.564
reading	79	4.144	0.525	5.035	0.275	3.982	0.518
hey	79	4.333	0.868	5.131	0.364	4.198	0.8
learn	79	4.289	0.809	5.017	0.247	4.15	0.763
plan	79	4.068	0.644	5	0.227	3.906	0.62
fine	79	4.124	0.581	5.037	0.28	3.976	0.544
move	79	4.352	0.684	5.064	0.299	4.178	0.637
worth	78	4.286	0.687	5.08	0.336	4.027	0.701
terms	78	4.168	0.644	5.037	0.28	4.082	0.637
weekend	78	4.152	0.56	5.013	0.229	3.88	0.571
[laughter-it]	78	4.649	0.88	5.298	0.431	4.423	0.882
unless	78	4.276	0.677	5.058	0.303	4.145	0.633
kid	77	4.213	0.677	5.05	0.286	4.002	0.617
trouble	77	4.234	0.565	5.1	0.337	4.151	0.615
shows	77	4.255	0.599	4.987	0.193	4.129	0.571
trees	77	4.219	0.648	5	0.217	4.101	0.592
dad	77	4.133	0.604	5.076	0.316	4.025	0.587
homes	77	4.26	0.622	4.99	0.207	4.066	0.595
interested	76	4.151	0.683	5.018	0.232	3.995	0.654
vacation	76	4.116	0.703	5.032	0.269	3.886	0.734
except	76	4.21	0.598	5.071	0.313	4.082	0.546
using	76	4.069	0.635	4.983	0.173	3.959	0.597

store	76	4.089	0.555	5.036	0.269	3.957	0.497
bye	76	4.096	0.831	5.25	0.425	3.963	0.713
ought	76	4.423	0.618	5.119	0.357	4.294	0.544
kept	75	4.198	0.605	5.075	0.315	4.101	0.558
television	75	4.119	0.705	5.085	0.332	3.927	0.655
ways	75	4.139	0.524	4.965	0.146	4.011	0.513
needs	75	4.009	0.791	5.01	0.238	3.955	0.623
yard	75	4.237	0.652	4.963	0.151	4.055	0.662
who's	75	4.21	0.531	5.035	0.258	4.071	0.485
families	74	4.149	0.601	5.051	0.284	4.018	0.553
th[e]-	74	4.024	0.644	5.067	0.318	4.014	0.648
starting	74	4.214	0.614	5.026	0.264	4.088	0.579
sitting	74	4.224	0.67	5.098	0.346	4.074	0.605
door	74	4.143	0.578	5.011	0.221	3.953	0.533
community	74	4.021	0.545	4.978	0.181	3.846	0.481
exercise	74	4.258	0.581	5.068	0.313	4.041	0.516
book	74	4.273	0.645	5.083	0.326	4.106	0.597
plastic	73	4.077	0.537	4.965	0.145	3.862	0.563
cards	73	4.137	0.667	5.037	0.283	3.962	0.591
felt	73	4.256	0.624	5.131	0.362	4.188	0.595
particular	73	4.194	0.579	4.97	0.152	4.025	0.511
themselves	73	4.204	0.659	5.003	0.236	4.012	0.63
fishing	73	4.235	0.676	5.04	0.279	4.057	0.615
phone	72	4.352	0.616	5.112	0.358	4.131	0.549
it'll	72	4.191	0.543	5.114	0.36	4.08	0.554
winter	72	4.136	0.535	5.067	0.309	3.986	0.487
students	72	4.14	0.59	4.96	0.121	3.945	0.605
everyone	72	4.181	0.604	5.076	0.311	4.032	0.538
behind	72	4.073	0.532	4.989	0.18	3.919	0.528
a[nd]-	72	4.102	0.753	5.202	0.408	4.009	0.69
wish	72	4.459	0.69	5.233	0.418	4.419	0.639
class	72	4.229	0.497	5	0.225	4.026	0.495
depends	71	4.1	0.589	5.066	0.314	4.106	0.583
men	71	4.247	0.529	5.072	0.32	4.118	0.522
programs	71	4.086	0.528	5.004	0.229	3.942	0.526
plus	71	4.17	0.583	5.102	0.34	4.136	0.627
hadn't	71	4.362	0.661	5.171	0.398	4.276	0.636
group	71	4.171	0.715	5.039	0.274	3.932	0.658
seventy	71	4.091	0.541	5.01	0.237	3.962	0.531
open	71	4.297	0.614	5.093	0.331	4.107	0.624
spending	71	4.276	0.585	4.987	0.189	4.069	0.6
you'll	71	4.176	0.69	5.131	0.364	4.176	0.659
law	70	4.222	0.558	4.997	0.219	4.074	0.556
[laughter-right]	70	4.231	0.971	5.396	0.457	4.164	0.975
afford	70	4.288	0.519	5.055	0.291	4.056	0.538
running	70	4.315	0.643	5.072	0.322	4.159	0.592
playing	70	4.215	0.567	5.088	0.327	4.027	0.577
late	70	4.238	0.613	5.026	0.268	4.064	0.53
hit	70	4.195	0.615	5.075	0.316	4.099	0.603
income	70	4.239	0.574	5.02	0.25	4.048	0.556
bunch	70	4.298	0.617	5.101	0.333	4.164	0.65
hate	69	4.371	0.657	5.172	0.387	4.24	0.6

across	69	4.292	0.549	4.966	0.423	4.105	0.526
somewhere	69	4.256	0.65	5.03	0.279	4.035	0.668
catch	69	4.202	0.669	5.076	0.324	4.091	0.586
turned	69	4.215	0.686	5.046	0.288	4.104	0.698
ah	69	3.929	0.711	5.136	0.371	3.8	0.724
dollar	69	4.28	0.506	5.06	0.286	4.116	0.502
finally	69	4.246	0.766	5.12	0.361	4.162	0.651
chance	69	4.184	0.517	5.038	0.271	3.975	0.546
goodness	69	3.951	0.569	5.044	0.487	4.022	0.548
sorry	68	4.163	0.798	5.078	0.329	4.033	0.777
medical	68	4.043	0.584	4.989	0.211	3.851	0.588
regular	67	4.206	0.611	5.014	0.246	3.976	0.607
tough	67	4.048	0.678	5.087	0.319	4.017	0.633
teachers	67	4.355	0.498	5.023	0.261	4.196	0.488
charge	67	4.201	0.641	4.99	0.2	4.057	0.599
level	67	4.271	0.443	4.976	0.167	4.056	0.504
about_1	67	4.306	0.686	5.049	0.292	4.153	0.58
throw	67	4.139	0.583	5.028	0.272	3.956	0.571
longer	67	4.119	0.64	5.015	0.252	3.941	0.55
golf	67	4.119	0.595	5.019	0.253	3.913	0.552
weren't	67	4.263	0.584	5.055	0.301	4.113	0.569
baby	67	4.158	0.715	5.047	0.3	3.903	0.643
team	66	4.165	0.657	5.013	0.246	3.961	0.663
grow	66	4.303	0.557	5.117	0.355	4.163	0.559
amazing	66	4.041	0.569	5.079	0.308	3.989	0.515
white	66	4.032	0.734	5.129	0.357	3.932	0.689
later	66	4.27	0.514	5.085	0.33	4.165	0.434
totally	66	4.045	0.621	5.052	0.293	3.867	0.531
woman	66	4.263	0.601	5.084	0.333	4.117	0.602
learned	66	4.384	0.556	5.066	0.307	4.24	0.589
information	66	4.2	0.614	5.035	0.271	3.976	0.603
restaurant	66	4.258	0.606	5.135	0.379	4.111	0.556
society	66	4.23	0.598	5.044	0.288	3.95	0.564
military	65	3.962	0.636	5	0.221	3.793	0.57
easier	65	4.086	0.544	5.021	0.264	3.897	0.523
ours	65	4.247	0.51	5.087	0.325	4.1	0.432
knows	65	4.207	0.494	5.078	0.317	4.085	0.479
dress	65	4.177	0.49	5.032	0.26	4.023	0.432
its	65	4.24	0.59	5.063	0.306	4.138	0.549
sell	65	4.339	0.629	5.066	0.311	4.13	0.673
particularly	65	4.152	0.563	5.082	0.319	4.008	0.523
[laughter-so]	65	4.532	1.088	5.306	0.589	4.373	1.037
punishment	64	3.959	0.616	4.958	0.12	3.877	0.581
cook	64	4.043	0.538	5.011	0.23	3.925	0.529
dinner	64	4.036	0.508	5.011	0.249	3.865	0.487
season	64	4.204	0.538	4.98	0.197	4.027	0.556
nineteen	64	4.175	0.553	5.052	0.292	3.987	0.465
power	63	4.106	0.527	4.994	0.215	3.915	0.54
consider	63	4.223	0.583	5.074	0.32	4.048	0.577
story	63	4.138	0.708	4.998	0.213	3.943	0.612
built	63	4.276	0.61	5.011	0.231	4.218	0.602
degree	63	4.074	0.577	4.976	0.153	3.86	0.535

realize	63	4.286	0.64	5.082	0.32	4.171	0.588
ninety	62	4.042	0.453	5.01	0.245	3.931	0.401
testing	62	4.034	0.597	4.95	0.111	3.817	0.553
near	62	4.083	0.452	5.04	0.275	3.893	0.435
beautiful	62	4.075	0.608	5.101	0.342	3.954	0.561
hold	61	4.483	0.731	5.165	0.379	4.404	0.674
feeling	61	4.216	0.519	5.076	0.309	4.095	0.511
buying	61	4.106	0.657	5.041	0.292	3.969	0.601
fast	61	4.179	0.605	4.999	0.227	3.927	0.823
forth	60	4.341	0.601	5.01	0.241	4.095	0.569
happy	60	4.196	0.599	5.098	0.348	4.044	0.608
choice	60	4.168	0.545	4.975	0.171	4.003	0.538
given	60	3.967	0.416	5.023	0.259	3.814	0.399
wants	60	4.229	0.562	5.075	0.318	4.008	0.611
bags	60	3.984	0.563	5.001	0.233	3.871	0.574
sixty	60	4.206	0.609	5.036	0.282	4.032	0.625
worse	60	4.162	0.529	5.022	0.262	3.967	0.513
single	60	4.23	0.513	5.089	0.332	4.088	0.507
send	60	4.199	0.588	5.105	0.342	4.074	0.551
become	60	4.071	0.564	4.952	0.099	3.936	0.544
brought	59	4.216	0.601	5.1	0.337	4.132	0.523
guns	59	4.032	0.547	5.016	0.26	3.887	0.54
cans	59	4.023	0.523	4.999	0.223	3.839	0.513
extra	59	4.189	0.672	5.013	0.251	4.008	0.572
cases	59	4.358	0.503	5.039	0.282	4.194	0.555
necessarily	59	4.109	0.566	5.057	0.307	3.97	0.577
[laughter-and]	59	4.814	1.018	5.329	0.445	4.611	0.994
tha[t]-	59	4.041	0.556	5.119	0.362	3.865	0.576
terrible	59	4.054	0.59	5.102	0.351	3.96	0.546
church	59	4.207	0.647	5.046	0.28	4.086	0.614
[laughter-that]	59	4.33	0.899	5.213	0.404	4.129	0.882
concerned	59	4.267	0.549	5.004	0.218	4.116	0.556
capital	59	4.322	0.69	5.013	0.253	4.148	0.57
scary	59	4.171	0.588	5.079	0.306	4.111	0.668
example	58	4.212	0.484	5.126	0.351	4.171	0.514
needed	58	4.154	0.611	5.046	0.284	4.028	0.571
generally	58	3.886	0.596	5.04	0.263	3.823	0.566
means	58	4.162	0.559	5.025	0.26	4.055	0.587
thank	58	4.454	0.725	5.438	0.45	4.367	0.704
sad	58	3.992	0.691	5.07	0.313	4.054	0.556
personal	58	4.167	0.839	5.013	0.236	4.023	0.754
period	58	4.047	0.609	5.053	0.298	3.914	0.627
newspaper	58	4.214	0.586	5.046	0.291	3.941	0.61
low	58	4.103	0.497	5.014	0.244	3.927	0.493
general	58	4.087	0.731	4.987	0.202	3.99	0.721
research	58	4.159	0.641	4.984	0.199	3.949	0.618
boys	58	4.21	0.654	5.007	0.234	4.007	0.687
huh-uh	58	3.854	0.663	5.048	0.292	3.765	0.52
perhaps	58	4.158	0.739	5.04	0.283	4.006	0.719
biggest	58	4.221	0.576	5.047	0.3	4.129	0.555
break	57	4.146	0.48	5.012	0.246	4.016	0.497
countries	57	4.191	0.617	5.045	0.283	4.023	0.629

judge	57	4.095	0.588	5	0.212	3.912	0.62
football	57	4.171	0.554	5.058	0.288	4.018	0.522
jobs	57	4.082	0.562	5.008	0.227	3.867	0.494
boat	57	4.194	0.603	5.021	0.252	4.002	0.545
clean	57	4.239	0.772	5.127	0.363	4.059	0.763
private	57	3.963	0.476	4.994	0.216	3.769	0.503
eighteen	56	4.013	0.553	5.084	0.335	3.882	0.508
test	56	4.204	0.587	5.03	0.271	4.032	0.507
younger	56	4.219	0.711	5.066	0.315	4.048	0.641
teacher	56	4.197	0.564	5.022	0.262	4.003	0.519
space	56	4.101	0.68	5.037	0.285	3.948	0.533
rate	56	4.23	0.648	5.034	0.272	4.067	0.667
brother	56	4.114	0.591	5.026	0.267	3.942	0.526
issue	56	4.148	0.724	5.057	0.298	4.032	0.635
special	56	4.154	0.677	5.068	0.31	4.023	0.635
social	55	4.233	0.591	5.027	0.25	4.103	0.507
figure	55	4.255	0.552	5.071	0.323	4.121	0.627
size	55	4.14	0.539	4.982	0.189	4.02	0.537
price	55	4.08	0.659	5.056	0.306	3.947	0.642
anywhere	55	4.07	0.692	5.05	0.293	3.81	0.666
quality	55	4.192	0.693	5.108	0.347	4.05	0.626
ended	55	4.271	0.45	5.046	0.285	4.196	0.511
lost	55	4.315	0.577	5.045	0.287	4.164	0.531
twice	54	4.181	0.483	4.975	0.178	4.013	0.436
subject	54	4.452	0.693	5.154	0.382	4.319	0.71
glass	54	3.893	0.667	4.994	0.197	3.836	0.629
caught	54	4.163	0.613	5.05	0.282	4.047	0.59
worry	54	4.253	0.63	5.064	0.31	3.98	0.666
decide	54	4.037	0.482	5.032	0.271	3.899	0.461
beginning	54	4.189	0.606	5.047	0.28	3.998	0.601
within	54	4.399	0.741	5.043	0.293	4.21	0.661
national	54	4.165	0.597	5.003	0.237	3.939	0.596
student	54	4.157	0.582	5.026	0.255	3.949	0.561
th[at]-	53	4.274	0.756	5.206	0.406	4.197	0.806
personally	53	4.23	0.709	5.07	0.312	4.231	0.727
father	53	4.22	0.672	5.072	0.307	4.07	0.568
black	53	4.137	0.542	5.071	0.305	4.028	0.542
awful	53	4.26	0.67	5.131	0.357	4.224	0.727
short	53	4.022	0.467	5.028	0.259	3.92	0.48
salary	53	4.069	0.615	5.018	0.236	3.879	0.654
likes	53	4.099	0.624	4.98	0.184	3.968	0.559
it'd	53	4.238	0.595	5.101	0.336	4.112	0.597
pets	52	4.062	0.618	4.979	0.175	3.82	0.616
save	52	4.135	0.553	5.108	0.346	4.001	0.518
crazy	52	4.118	0.512	5.044	0.297	3.904	0.486
vote	52	4.339	0.625	5.039	0.277	4.071	0.632
teach	52	4.266	0.519	4.996	0.217	4.06	0.489
mom	52	4.211	0.675	5.113	0.353	4.067	0.632
books	52	4.309	0.712	4.977	0.182	4.104	0.75
newspapers	51	4.047	0.557	4.97	0.163	3.864	0.561
trip	51	4.198	0.635	4.994	0.206	3.998	0.57
trial	51	4.075	0.644	4.991	0.219	3.826	0.659

costs	51	4.308	0.649	5.077	0.323	4.079	0.677
neighborhood	51	4.162	0.496	4.962	0.139	3.918	0.433
bigger	51	4.063	0.424	5.015	0.248	3.948	0.494
pollution	51	4.13	0.655	5.049	0.298	3.989	0.615
center	51	4.289	0.737	4.998	0.221	4.055	0.709
lose	51	4.33	0.626	5.035	0.276	4.132	0.552
teaching	50	4.071	0.548	5.011	0.234	3.856	0.473
mess	50	4.164	0.917	5.043	0.289	3.977	0.804
minute	50	4.344	0.845	5.06	0.305	4.242	0.786
gee	50	4.225	0.892	5.058	0.304	4.077	0.858
stand	50	4.205	0.538	5.067	0.325	4.161	0.57
waiting	50	4.449	0.553	5.018	0.254	4.235	0.548
changes	49	4.201	0.591	5.056	0.3	3.968	0.551
asked	49	4.333	0.554	5.075	0.325	4.161	0.547
suppose	49	4.235	0.707	5.183	0.39	4.16	0.681
magazines	49	4.271	0.686	5.067	0.306	4.049	0.678
guilty	49	4.131	0.485	5.018	0.254	3.92	0.459
term	49	4.236	0.576	5.029	0.264	4.006	0.605
grass	49	4.163	0.677	5.037	0.268	4.032	0.697
glad	49	4.257	0.765	5.222	0.409	4.215	0.777
soon	49	4.116	0.553	5.021	0.26	4.029	0.535
ha[ve]-	49	4.459	0.661	5.228	0.418	4.364	0.648
wh[at]-	49	4.267	0.772	5.178	0.399	4.239	0.793
rid	49	4.346	0.695	5.146	0.381	4.179	0.707
garden	49	4.06	0.542	4.98	0.186	3.938	0.562
stop	49	4.205	0.666	5.087	0.317	4.041	0.649
sister	49	4.183	0.437	5.077	0.326	3.998	0.451
we[ll]-	49	4.041	0.764	5.151	0.381	4.056	0.679
types	48	4.206	0.57	5.013	0.254	4.068	0.567
driving	48	4.109	0.577	5.019	0.257	3.904	0.604
environment	48	4.015	0.469	4.975	0.187	3.833	0.485
keeping	48	4.235	0.492	5.014	0.232	4.098	0.454
ooh	48	3.747	0.706	5.081	0.326	3.685	0.635
anyone	48	4.056	0.602	5.016	0.254	3.853	0.618
building	48	4.036	0.494	4.957	0.132	3.856	0.457
attention	48	4.35	0.494	5.027	0.253	4.024	0.584
higher	48	4.295	0.723	5.092	0.348	4.076	0.687
telling	48	4.152	0.585	5.063	0.308	4.003	0.566
cause	48	4.24	0.669	5.178	0.4	4.144	0.74
watched	48	4.106	0.443	5.205	0.406	4.013	0.436
[laughter-oh]	48	4.687	1.087	5.364	0.452	4.528	1.043
north	48	3.939	0.558	5.013	0.253	3.706	0.489
recycle	47	4.062	0.544	5.02	0.252	3.928	0.517
carry	47	4.128	0.63	4.981	0.185	3.914	0.533
support	47	4.168	0.629	5.044	0.284	4.091	0.595
whenever	47	4.067	0.631	5.044	0.277	4.069	0.64
main	47	4.355	0.677	5.025	0.259	4.194	0.622
rent	47	4.285	0.58	5.088	0.333	4.132	0.572
normal	47	4.095	0.498	4.942	0.023	3.997	0.412
eventually	47	4.08	0.488	5.031	0.272	3.976	0.587
doctor	47	4.073	0.53	5.004	0.225	3.85	0.541
oil	47	3.964	0.604	5.025	0.262	3.816	0.62

seemed	47	4.337	0.695	5.13	0.351	4.29	0.745
handle	47	4.306	0.598	5.029	0.272	4.074	0.565
calling	47	4.249	0.625	5.207	0.412	4.061	0.583
parts	47	4.205	0.602	5.007	0.23	3.978	0.56
word	46	4.115	0.545	4.969	0.157	4.044	0.518
keeps	46	4.091	0.593	5.12	0.365	3.985	0.541
serve	46	4.277	0.828	5.162	0.366	4.121	0.852
head	46	4.285	0.654	5.092	0.326	3.98	0.646
luck	46	4.487	0.678	5.252	0.429	4.342	0.656
pull	46	4.159	0.767	5.089	0.335	4.052	0.716
apartment	46	4.425	0.496	5.056	0.288	4.18	0.487
fan	46	4.098	0.476	5.014	0.243	3.798	0.639
houses	46	4.046	0.467	5.005	0.225	3.955	0.459
gas	46	4.173	0.525	4.966	0.142	3.956	0.429
itself	46	4.126	0.572	5.005	0.233	3.949	0.571
obviously	46	4.147	0.595	5.07	0.316	3.961	0.544
picked	46	4.177	0.537	5.013	0.235	4.041	0.484
based	46	4.253	0.636	5.107	0.345	4.121	0.591
lucky	46	4.173	0.551	5.076	0.296	4.055	0.412
w[ell]-	46	4.312	0.79	5.227	0.415	4.216	0.804
basis	45	4.308	0.511	5.07	0.315	4.007	0.508
played	45	4.251	0.71	5.003	0.234	4.094	0.703
everyday	45	4.166	0.624	5.019	0.264	3.965	0.573
calls	45	4.254	0.618	5.065	0.306	4.027	0.698
strange	45	4.073	0.717	5.131	0.361	3.993	0.712
restaurants	45	4.243	0.471	5.078	0.321	3.989	0.512
bill	45	4.324	0.753	5.042	0.27	4.073	0.798
privacy	45	4.125	0.612	5.007	0.236	3.843	0.59
noticed	45	4.215	0.677	5.072	0.326	4.18	0.661
order	44	4.125	0.45	5	0.215	3.925	0.471
crimes	44	4.038	0.583	4.98	0.195	3.91	0.533
camp	44	4.131	0.744	4.98	0.195	3.945	0.69
stick	44	4.251	0.48	5.027	0.269	4.088	0.536
sports	44	4.278	0.646	5.02	0.266	4.047	0.532
girl	44	4.101	0.775	5.007	0.234	3.904	0.655
activities	44	3.987	0.455	5.02	0.239	3.814	0.446
per	44	4.221	0.668	4.985	0.195	4.082	0.594
similar	44	4.346	0.701	5.038	0.26	4.219	0.642
willing	44	4.265	0.615	4.988	0.196	4.086	0.59
completely	44	4.043	0.558	5.037	0.275	3.834	0.602
chicken	44	3.941	0.478	4.986	0.196	3.805	0.444
shoot	43	4.248	0.66	5.041	0.276	4.121	0.617
answer	43	4.3	0.632	5.107	0.338	4.103	0.672
rain	43	4.145	0.621	5.014	0.232	3.981	0.573
huge	43	4.225	0.731	5.003	0.211	4.132	0.571
parent	43	4.152	0.763	5.046	0.294	3.963	0.734
kill	43	4.11	0.85	5.07	0.296	3.981	0.788
stayed	43	4.222	0.707	5.123	0.354	4.077	0.668
i[t's]-	43	4.082	0.828	5.11	0.342	3.998	0.737
cover	43	4.059	0.645	5.036	0.25	3.939	0.649
retired	43	4.263	0.429	5.094	0.33	4.094	0.463
mountains	43	4.172	0.513	5.027	0.263	4.038	0.528

bills	43	4.1	0.573	5.001	0.236	3.893	0.511
hi	43	3.961	0.753	5.114	0.351	3.907	0.79
party	43	4.042	0.656	5.017	0.254	3.924	0.62
forget	42	4.295	0.805	5.177	0.397	4.13	0.78
street	42	4.296	0.511	5.054	0.304	3.974	0.46
third	42	4.176	0.636	4.981	0.199	4.067	0.545
tree	42	4.153	0.592	4.995	0.214	4.048	0.586
fair	42	4.093	0.566	5.176	0.387	3.965	0.615
write	42	4.355	0.69	4.986	0.199	4.165	0.663
unfortunately	42	4.192	0.677	5.074	0.312	4.045	0.624
penalty	42	4.146	0.559	4.971	0.161	4.01	0.566
seeing	42	4.153	0.61	5.02	0.245	3.985	0.538
eleven	42	4.154	0.514	5.016	0.239	3.968	0.483
smaller	42	4.02	0.423	5.018	0.245	3.909	0.502
giving	42	4.169	0.482	4.981	0.199	3.976	0.361
process	42	4.014	0.591	5.025	0.274	3.848	0.611
lake	42	4.416	0.637	4.971	0.162	4.199	0.574
sick	42	4.112	0.644	5.067	0.314	3.893	0.547
excellent	41	4.171	0.555	5.017	0.247	3.937	0.557
grade	41	4.02	0.612	5.03	0.25	3.836	0.624
girls	41	4.37	0.765	4.983	0.53	4.132	0.69
sound	41	4.506	0.541	5.155	0.376	4.399	0.617
died	41	4.072	0.631	5.043	0.289	3.881	0.56
serious	41	4.172	0.552	5.045	0.26	4.07	0.497
weird	41	4.112	0.588	5.016	0.245	4.028	0.515
rock	41	4.268	0.855	5.096	0.337	4.074	0.774
lawn	41	4.126	0.523	5.115	0.364	3.968	0.436
wood	41	4.129	0.624	5.004	0.238	3.893	0.589
color	41	4.126	0.533	5.076	0.3	3.9	0.517
paint	41	4.003	0.604	5.022	0.256	3.861	0.551
decision	41	3.921	0.557	5.015	0.249	3.89	0.556
covered	41	4.375	0.731	5.091	0.339	4.238	0.76
apparently	41	4.127	0.521	5.156	0.379	4.012	0.663
view	41	4.128	0.574	5.013	0.208	4.019	0.561
he'll	40	4.206	0.525	5.123	0.356	4.159	0.537
savings	40	4.048	0.544	4.986	0.203	3.916	0.549
market	40	4.371	0.711	5.077	0.314	4.148	0.669
heart	40	4.477	0.613	5.173	0.385	4.329	0.587
he'd	40	4.269	0.609	5.084	0.329	4.222	0.667
shouldn't	40	4.328	0.661	5.118	0.362	4.235	0.672
gives	40	4.137	0.618	5.141	0.368	4.046	0.616
cash	40	4.202	0.537	5.007	0.246	4.046	0.57
metric	40	4.359	0.652	5.074	0.309	4.163	0.653
respect	40	4.184	0.456	5.098	0.351	4.026	0.465
coverage	40	4.173	0.566	4.993	0.208	4.022	0.574
states	40	4.126	0.62	5.03	0.281	3.915	0.65
build	40	4.19	0.581	5.08	0.311	4.081	0.607
raise	40	4.415	0.616	5.088	0.316	4.185	0.656
y[eah]-	40	3.956	0.64	5.313	0.443	4.009	0.677
helps	40	4.164	0.743	5.198	0.415	4.178	0.704
[laughter-the]	40	5.246	1.102	5.409	0.445	4.98	1.079
afraid	40	4.305	0.591	5.095	0.338	4.222	0.509

available	40	4.245	0.503	5.03	0.26	4.046	0.554
w[e]-	40	4.187	0.757	5.054	0.301	4.156	0.773

Curriculum Vitae

Shreyas A. Karkhedkar was born in Nagpur, India on 21st of April, 1986 as first of two children of Ashok M. Karkhedkar and Aditi A. Karkhedkar. Shreyas began pursuing his combined Bachelors and Masters degree in Computer Science and Engineering at the Indian Institute of Technology (IIT), Kharagpur in 2004. In 2009, after graduating from IIT, Shreyas began his doctoral studies in Computer Science at the University of Texas at El Paso under the guidance of Dr. Nigel Ward.

While pursuing his Doctoral degree, Shreyas interned at Google Inc. in the summers of 2011 and 2012 with the Google AdSense and Google Analytics teams. In May 2013, Shreyas became another doctorate in his family.

Shreyas will be working full-time at Google with the Google Analytics team based in Mountain View, California.