Improving Speech Recognition Accuracy for Clinical Conversations

by

Burkay Gur

S.B., Electrical Engineering, MIT, 2011

Submitted to the Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Electrical Engineering and Computer Science
at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2012

Author . . . . . .
Department of Electrical Engineering and Computer Science
May 21, 2012

Certified by . . . . . .
Peter Szolovits
Professor of Computer Science and Engineering
Thesis Supervisor

Certified by . . . . . .
Dennis M. Freeman
Chairman, Master of Engineering Thesis Committee

Improving Speech Recognition Accuracy for Clinical Conversations

by

Burkay Gur

Submitted to the Department of Electrical Engineering and Computer Science
on May 21, 2012, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

Accurate and comprehensive data form the lifeblood of health care. Unfortunately, there is much evidence that current data collection methods sometimes fail. Our hypothesis is that it should be possible to improve the thoroughness and quality of information gathered through clinical encounters by developing a computer system that (a) listens to a conversation between a patient and a provider, (b) uses automatic speech recognition technology to transcribe that conversation to text, (c) applies natural language processing methods to extract the important clinical facts from the conversation, (d) presents this information in real time to the participants, permitting correction of errors in understanding, and (e) organizes those facts into an encounter note that could serve as a first draft of the note produces by the clinician. In this thesis, we present our attempts to measure the performances of two state-of-the-art automatic speech recognizers (ASRs) for the task of transcribing clinical conversations, and explore the potential ways of optimizing these software packages for the specific task. In the course of this thesis, we have (1) introduced a new method for quantitatively measuring the difference between two language models and showed that conversational and dictational speech have different underlying language models, (2) measured the perplexity of clinical conversations and dictations and shown that spontaneous speech has a higher perplexity than dictational speech, (3) improved speech recognition accuracy by language adaptation using a conversational corpus, and (4) introduced a fast and simple algorithm for cross talk elimination in two speaker settings.

Acknowledgments

It would not have been possible for me to write this thesis without the help of the great people around me. Even though I can only particularly mention some of them here, I remain grateful to all who have supported me in this arduous journey.

First and foremost, I would like to express my deepest gratitude to my thesis advisor Peter Szolovits. His invaluable insight, suggestions and encouragement made this thesis possible. One could not ask for a better guidance, and for that I thank him.

I would like to acknowledge the National Library of Medicine Grant R01 LM009723. I would also like to thank Nuance Communications for providing their commercial software in support of this project and SRI International for providing an evaluation license, the importance of which cannot be emphasized enough. I would like to particularly thank Tom Lynch from Nuance for his guidance in getting technical support.

I cannot thank enough to my dear friends Umut Varolgunes, Jocelyn Fuentes and Ameya Shroff for simply being there. Knowing such wonderful people and spending time with them through this journey has been an indescribable pleasure. I will miss you dearly.

I would also like to thank Firat Ileri, Yunus Sasmaz, Kaan Karamanci and Rahul Shroff for they have taught me a great deal during my first times at MIT and helped me pave the way to this successful conclusion.

Finally, I would like to express my gratitude to my grandmother Hanife for helping to raise me; and for their unconditional love and support, I am eternally thankful to my mom Leyla, my dad Hasan, and my little brother Umut. This thesis is dedicated to them.

Contents

Chapter 1

Introduction

1.1 Vision

Accurate and comprehensive data form the lifeblood of health care. Such data are needed to allow clinicians to understand the patient and to deliver the best care. They also provide an opportunity for administrators to assess the quality of care being delivered and researchers to learn the natural course of diseases, the accuracy of tests, and the effectiveness of therapies. Collection of data during clinical encounters should also be efficient so that it does not slow down the care process. Unfortunately, there is much evidence that current data collection methods sometimes fail; information elicited during an encounter may fail to be recorded or may be recorded incorrectly. Such omissions or errors undermine not only clinical care but also secondary uses of such data in quality improvement, public health and research.

One way to tackle this problem is to computerize health records in order to "avoid dangerous medical mistakes, reduce costs, and improve care", as former

President George W. Bush mentioned and set the goal to do so by 2014. Among the goals of the proposed healthcare reform of 2010 was the transition to electronic health records to enable better, more consistent care, analysis of spending, improvement of process and enhanced research.

1.2 How Much Data is Lost?

Clinical data are collected in many ways in the care process, and with the popularization of electronic medical records, we anticipate a day in which all these data become accessible any time they are needed. Data such as laboratory measurements are now routinely collected automatically by instruments, as are some data from bedside monitors and imaging machinery. Nevertheless, much of the data in clinical care is still collected through the direct observation of clinicians and recorded in textual reports by nurses and doctors. Other significant data about a patient, such as his or her family history, the history of the present illness (what brought the patient to the doctor), a description of what medicines the patient is taking, recent medical visits, tests and treatments at other institutions, and notable environmental exposures, subjective symptoms and self-observations are ordinarily transmitted to the nurse or doctor through conversation during an office or clinic visit.

Data acquired from these conversations may, however, be lost or incorrectly recorded. For example, most doctors are accustomed to writing or dictating notes on a patient visit after the patient has left. Thus, the note-taking is prone to misunderstanding or mishearing the patient, forgetting chunks of the information the patient presents, skipping over a detail that might be crucial later on and so on. Furthermore, if such records are not kept in a database so that the care provider could reference later on, healthcare system would suffer from more inefficiency. Research suggests that the theory is in fact correct. *The Computer-Based Patient Record: An Essential Technology for Health Care,* a 1991 report by The Institute of Medicine [6] shows how medical records fail to reflect patients' histories accurately. Among the studies cited in the report are a 1975 study that compared tape-recorded conversations with patient records and found significant omissions in crucial categories such as reason of visit and degree of disability, and a 1981 study where an independent observer took notes as well as the doctor, and comparison of those later revealed a match of 71%-73% for diagnosis, tests and information related to the current illness, and even less success for medical history.

Chapter 2

Proposed Solution

2.1 The Fairwitness Project

Our hypothesis is that it should be possible to improve the thoroughness and quality of information gathered through clinical encounters by developing a computer system that (a) listens to a conversation between a patient and a provider, (b) uses automatic speech recognition technology to transcribe that conversation to text, (c) applies natural language processing methods to extract the important clinical facts from the conversation, (d) presents this information in real time to the participants, permitting correction of errors in understanding, and (e) organizes those facts into an encounter note that could serve as a first draft of the note produces by the clinician. We named our planned system Fairwitness, after the profession invented by Robert Heinlein in his novel *Stranger in a Strange Land*; the term defines an individual trained to observe events and report exactly what he or she sees and hears, making no extrapolations or assumptions. That broadly defines the goal of the project, a

computerized witness to accurately capture the primary clinical data real time, and provide feedback on the fly for corrections via a dynamic user interface. Figure 1 outlines the structure of Fairwitness.

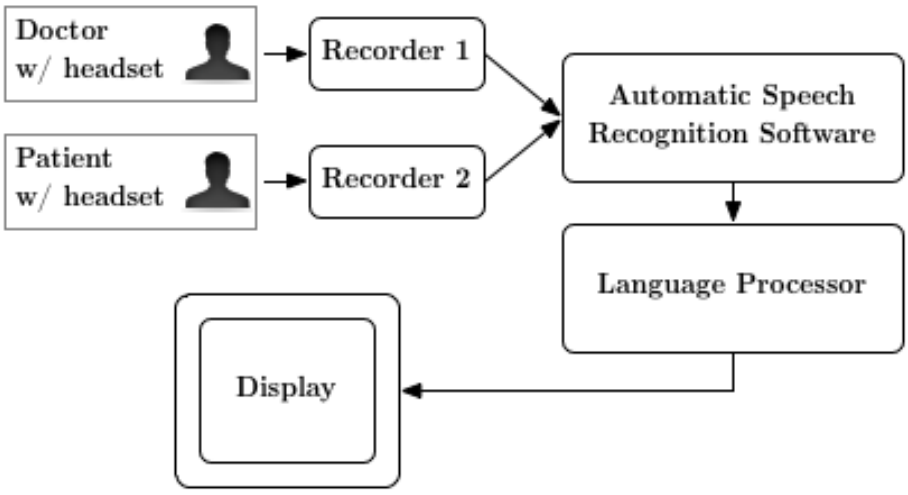

Figure 1: The structure of the Fairwitness Project

There are two components of such a system, namely speech to text and information extraction. The details of the workings of the latter are beyond the scope of this thesis. However, to briefly explain, the doctoral thesis by Christina Sauper from MIT's Natural Language Processing group uses text summarization algorithms [23] to form the backbone of the semantic analysis component of the Fairwitness project. The task of the speech to text component, which is the main focus of this thesis, is to listen to the conversation and transcribe it most accurately. There currently are several computer programs to achieve this task for a single-speaker setting, yet such technology does not exist for a multiple-

speaker setting. The approach is to modify the current single-speaker programs so that they can conserve their rate of accuracy in such settings and to tackle any other challenges that exist.

2.2 Technical Challenges

There are currently many challenges to implementing such a fully computerized environment even at the speech recognition stage due to the limited capabilities of current software technology. Such challenges can be summarized as follows:

Single vs. multiple speakers: As mentioned above, the automatic speech recognition software (ASR) can only listen to and analyze single speaker at a time yet the task at hand is to record and transcribe a conversation.

Trained vs. untrained speakers: Some ASRs utilize training where the individual reads chunks of text into the system so that it allows the ASR system to adapt its speech model to the speaking patterns and accent of the particular speaker. This is usually a long and elaborate process that would be difficult to implement in a hospital setting, thus the ASR at hand should be capable of providing results without any training.

Quiet vs. non-quiet backgrounds: The ASRs work better in a quiet environment because background noise could confuse the software. It is natural

to expect a certain level of background noise for two reasons: The conversation takes place in a hospital, which tends to be noisy, and in a conversational setting, while one speaker is speaking the other might not remain completely quiet.

Spoken language vs. dictated language: The ASRs are built to understand a speaker who is speaking directly into the system, yet in a conversational setting, the speakers are not going to dictate into the microphone and the phrases are not going to be plain and concise. Such lack of speaker collaboration could negatively affect the accuracy of the software.

In order to achieve our main goal, that is to engineer current systems so that they can handle this type of data, we focused on several technical questions:

1. We have developed formal methods to characterize the differences between the conversational language and the dictated language. A language modeling perspective revealed information about the differences of these two styles of languages.
2. We compared the performance of two well-known ASR systems, Dragon Naturally Speaking by Nuance (DNS '11) and SRI DynaSpeak. These programs also were optimized via customization for the specific context, i.e. doctor-patient medical conversations.

2.3 Related Work

The problems associated with manual data collection in clinical settings are under serious consideration. There have been several attempts in transcribing clinical conversations to minimize the amount of information loss. One of these attempts is the 1992 study by Fagan et al. "Q-MED: A Spoken-Language System to Conduct Medical Interviews" [18]. Much like Fairwitness, Q-MED is a system that facilitates medical interviews by making use of early ASRs. Even though the major problem they had was the inaccessibility of real data, we should also consider the fact that the ASRs have evolved exceptionally since then. A newer study conducted by Zafar et al. in 1999, "Continuous Speech Recognition for Clinicians" [19] focuses on getting doctors to use dictational speech by making use of speaker-trained ASRs. Although they obtained accurate transcription results, they interfere with the natural interaction between the doctor and the patient, by requiring the doctor to use a dictational tone. Overall, the research is particularly impeded by the problems that (1) the data are private and sensitive, so it has been difficult to get the broad research community involved and (2) the ASRs are trained to handle dictational data, which interferes with the natural flow of a clinical visit.

Chapter 3

Data Collection

3.1 Source of Speech Data

The pilot study for the project has been conducted at Children's Hospital Boston (CHB) Pediatric Environmental Health Clinic with Dr. Alan Woolf's collaboration. Dr. Woolf's expertise focuses on environmental health, and the vast majority of the cases seen in his clinic involve childhood lead poisonings. The main goal of the pilot study was to record 200 conversations and thus to collect data for system performance measurement and future processing and optimization tasks. So far, we have been able to collect 130 conversations. The patients have all consented to have their conversations used in research and our use of the data is approved by both CHB and MIT Institutional Review Boards (IRBs).

3.2 Recording Technology

For the recordings of the conversations, the two microphones we initially used were Andrea NC-91s, which were packaged with DNS '10 (though we subsequently used DNS '11 in our experiments). These were connected to two Sony MX20 recorders. With these, we initially encountered the problem that the other side of each conversation was faintly recorded in both recordings. On the recommendation of audio professionals, the microphones and the recorders were later replaced by Sennheiser ME3s, and Philips Pocket Memo 9600 Series, which were thought to improve on this problem. ASRs are optimized for single speakers. By recording the conversations through two different recorders, the speech recognition software should be able to interpret the utterances of each speaker in a conversation independently. Using two microphones has brought us an advantage for feeding each speaker to the ASR system individually, but that also introduced the crosstalk artifact. For example, when the doctor is speaking, the microphone attached to the mother picks up the doctor's voice and vice versa. Another problem is that the two separate recordings coming from the same conversation are not perfectly aligned, because the parties press the record buttons at different times. The lack of such alignment causes problems during (1) potential speaker segmentation efforts and (2) hand transcriptions of these conversations, which account for the gold standard data.

3.3 On-site Equipment

Our on-site equipment to store and analyze data is a Linux machine with an Intel Xeon 8-core 2.53 GHz processor and 12 GB RAM, and a Windows XP machine with an Intel Pentium 4 3.4 GHz processor and 3 GB RAM to run the ASRs. These machines are behind the CHB firewall to protect patient data, but accessible to us via VPN.

Protecting the privacy of patient data imposed a number of requirements on our project that slowed its progress significantly. Our protocol requires that patient data must be kept on hospital computers. The need to access data remotely impeded our process and made the availability of the data and performance of the project dependent on outside factors such as network interruptions and accidental powering down of our machines. This also required many visits to the hospital, to transfer data from the recorders to our computers and to correct the unanticipated practical impediments. Additionally, the Institutional Review Board (IRB) approval that was necessary to begin collecting data and to start working on its interpretation took an unexpectedly long time. Early in the project, we also needed multiple training sessions with the clinicians to acquaint them with the recording equipment and computer tools that were novel to them. Finally the need to maintain patient privacy has made it impossible for us to share data with outside researchers.

3.4 Gold Standard

As part of our research protocol, we hired the company Breitner Tanscription Services, Inc., to perform manual transcription of the first 66 our (currently 130) recordings of conversation. Breitner is used by Children's Hospital to transcribe doctors' dictations, so they are familiar with clinical language and have agreements in place to protect patient confidentiality. These transcriptions form the Gold Standard for our measurement of the accuracy of automatic speech transcription. The conversations were transcribed by multiple people, in different styles in terms of punctuations, paragraphs and data formats and thus had to be standardized. In order to achieve that standardization of the data, we performed post-processing such as, breaking the transcripts into its speakers, tokenization, removal of punctuation, and elimination of unnecessary explanations.

3.5 Dictational Corpus

In order to compare the difference between conversational speech and dictated speech, another set of data was necessary. Thus, dictated clinical notes by the doctors were also included in the dataset. These are the notes dictated by the doctor, and later hand-transcribed, after the patient is dismissed from the visit. We obtained permission to use dictated notes from the PEHC for up to 300 patients. Some, but not most, of these manually transcribed notes overlapped with the recorded conversations.

Chapter 4

Experiments

In this chapter, we present our attempts to measure the performances of two state-of-the-art ASRs for the task of transcribing clinical conversations, and explore the potential ways of optimizing these software packages for the specific task. We conducted five experiments:

1. We measured the baseline performance of Dragon Naturally Speaking '11 (DNS'11) by Nuance and DynaSpeak by the Stanford Research Institute (SRI) using their initial default settings. We have selected DNS '11 and SRI DynaSpeak, because of the high accuracy reported both commercially and in the literature.
2. We quantified the difference between two language models for conversations and dictations, by proposing a novel method.
3. We optimized the language models of DNS '11 and SRI DynaSpeak.
4. We calculated the perplexity of two language models derived from conversations and dictations.

5. We processed the input audio streams to eliminate unwanted background interferences caused by noise and the second speaker.

4.1 Choosing the Best ASR Default Model

In this experiment, we aim to see how the out-of-the-box models of DNS '11 and SRI DynaSpeak perform in transcribing clinical conversations. By seeing the performances of the default models, we can decide the type of errors that these ASRs make, and conduct necessary improvements.

Currently, the standard metric for measuring ASR performance is Word Error Rate (WER). Word Error Rate is a metric based on the Levenshtein distance, which is normally defined as the letter level "edit distance", but for our application applied on word level. Given a reference text as the gold standard, WER calculates the error percentage of a hypothesized text. It is given by this formula

$$WER = \frac{S + D + I}{N}$$

where S is the number of substitutions, D is the number of deletions, I is the number of insertions, C is the correctly identified words in the hypothesis text, and N is the number of words in the reference. Using dynamic programming, the words in both texts are aligned and types of errors are marked as below:

REF:	I	know	if	anyone	HAS	done	anything	**
HYP:	DON'T	know	if	anyone	***	done	anything	HE
Eval:	S	C	C	C	D	C	C	I

For the word alignment and WER calculation, we used the SCLITE package by the National Institute of Standards and Technology (NIST).

4.1.1 DNS '11 Default Model

First, we measure the WERs resulting from the transcriptions of the conversations by the generic DNS '11 user model involving no training of the speakers. Although we might have achieved better results by having each patient and doctor train DNS '11 to specialize the phonetic model to his or her voice, this proved impractical because it would have required patients to spend 30 minutes training the system before seeing their doctor. Table 1 shows an excerpt of the three highest, medium and lowest WERs for illustrative conversations from among our set of recordings.

Document	Correct (%)	Substitution (%)	Deletion (%)	Insertion (%)	WER
100329.p	18.7	80.4	0.9	171.3	252.6
090624.p	29.6	63.9	6.5	115.9	186.3
090624.d	17.9	80.7	1.5	92.3	174.4
090518.d	49.2	36.7	14.1	23.7	74.5
090902.p	54.3	36.2	9.5	28.7	74.4
090406.d	33.9	41.9	24.2	7.2	73.2
100407.d	66.1	20.4	13.5	8.3	42.2
090921.d	63.3	23.3	13.4	5.2	41.9
100303.d	67	20.4	12.6	4.9	37.9

Table 1: Word error rate statistics by non-trained DNS '11 default model. Documents ending with a ".p" and ".d" represent transcriptions coming from patient and doctor recordings respectively.

4.1.2 SRI DynaSpeak Default Model

Next, we measure the WERs introduced by SRI DynaSpeak. Due to frequent releases of updated models, we are only provided by the 2.10.2012 generic model as an evaluation version. SRI DynaSpeak is designed to work with short utterances of speech, such as single words, word groups or sentences. Therefore, we need to slice an input audio file into chunks without losing words.

For this task, we use the Sound eXchange (SoX) command line utility that provides various audio-processing tools, including a silence detector. Using the "silence" command we can slice an audio file where the average amplitude drops below an arbitrary percentage of the loudest point in the file. In this case, we have found 2% as a good cutoff point. There are two more parameters, attack and decay. Attack is the duration that the average has to be higher than 2% of

the highest amplitude in order to start the slicing. 0.1 seconds is found to be a good attack time. Decay is the duration that the average has to be lower than 2% of the highest amplitude in order to end the slicing. 0.2 seconds is found to be a good decay time. Figure 2 explains the attack and decay phenomena.

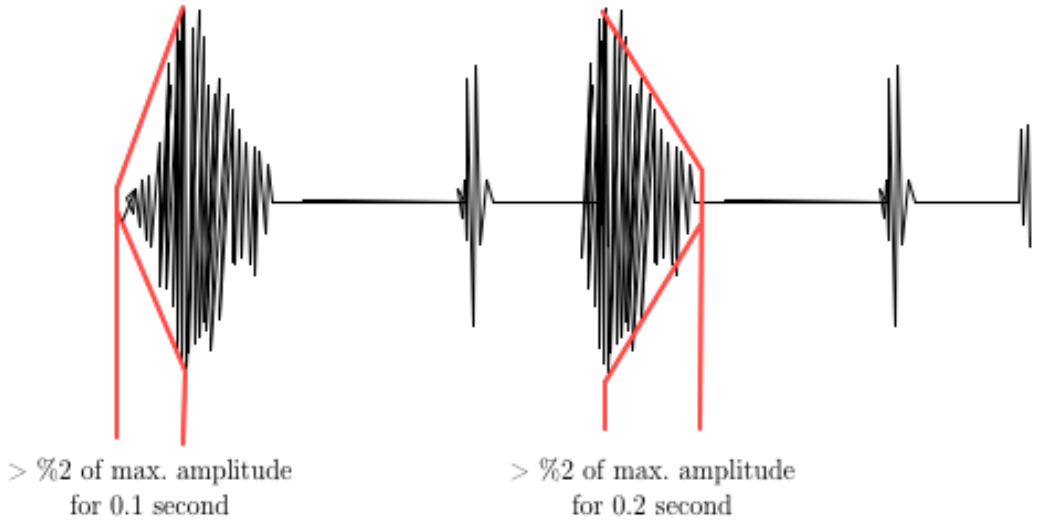

Figure 2: Attack/Decay phenomena. The recording will be sliced starting from the first marked region until the second marked region.

Using SoX's "newfile" command, a given audio file is split into smaller chunks to be fed into SRI DynaSpeak. Table 2 shows the WER statistics achieved by SRI DynaSpeak.

Document	Correct (%)	Substitution (%)	Deletion (%)	Insertion (%)	WER
100329.p	14.5	85.1	0.4	137	222.5
090624.p	22.4	72.7	4.9	46.7	124.3
090624.d	16.4	82.5	1.1	96.6	180.2
090518.d	39.6	41.9	18.5	8.5	68.9
090902.p	27.3	56.8	15.9	8.4	81.1
090406.d	40.9	40.1	19	8	67.1
100407.d	32.5	37.2	30.3	5.2	72.7
090921.d	45.4	32	22.6	4	58.6
100303.d	41	35.4	23.6	4.3	63.3

Table 2: Word error rate statistics by SRI DynaSpeak default model

These results are much higher than the advertised WERs of DNS '11 and SRI DynaSpeak, in fact, some WERs are higher than 100%. WER is measured by dividing the number of errors by the total number of words in the reference text, and the number of total errors can be greater than the total number of words in the reference text due to the fact that we count insertions as errors, which have no corresponding words in the reference text. Moreover, a manual investigation shows that the gold standard documents do not fully reflect the actual conversations, i.e. there are a lot of missing utterances. This is another reason why the WERs are very high as this leads to an inflated number of insertions.

4.2 Language Model Difference between Dictation and Conversation

In our second experiment, we aim to show that the language models for conversational (spontaneous) and dictational speech are different in their nature. Therefore, one of the explanations why the ASRs are performing so poorly might be the difference of these two types of speech, because we know that the DNS '11 and SRI DynaSpeak language models are both trained on dictational speech. We compare the conversational language model that we derived from the gold standard transcriptions with the language model derived from the hand transcribed dictations that the doctor records after the patient is dismissed. We do this comparison by using two different techniques: (1) by comparing the frequency distribution of word N-grams in the two types of text, and (2) by comparing the linguistic perplexity of the two. Since both ASRs do poorly on transcribing conversations, we expect that both analyses will come to similar conclusions showing that the language models for dictation differ significantly from those for conversation.

4.2.1 Finding a Metric

How do we measure the difference between two language models? Various studies focus on using average length, vocabulary coverage, term frequency and

inverted document frequency, and entropy analysis [2]. Since we know that the majority of ASRs are built on Hidden Markov Models (HMM) to represent the sequence of words and sounds in speech, we can devise a more relevant measure. ASR systems that perform speech-to-text transcription employ two major models: an acoustic model and a language model. The acoustic model uses a collection of audio features, typically in an HMM to represent the sequence of such features that are likely to be heard when a particular word is articulated. This model is useful for finding out what a given waveform could potentially correspond to in terms of phonemes. The language model contains information about the likelihood of which words will appear in what context. Therefore, the combination of an acoustic model and a language model returns the most likely phrase that the software thinks is being said by the speaker. Given an acoustical cue, the probability of a word sequence is P(W|A). By the Bayes rule:

$$P(W|A) = P(A|W) * P(W) / P(A).$$

Because P(A) is constant across a given acoustical utterance, we can compare the term P(A|W) * P(W) for different possible word sequences and pick the one maximizing this probability. In this case, P(A|W) is obtained from the acoustic model and P(W) is obtained from the language model. P(W) is equal to $P(w_1,$

$w_2,...,w_n$), where the w_i are the words in the sequence. Applying the chain rule yields:

$$P(w_1, w_2,...,w_n) = P(w_1) \cdot P(w_2|w_1) \cdot P(w_3|w_1, w_2) .... P(w_n| w_1, w_2,...,w_{n-1})$$

If we try to compute $P(w_1, w_2,...,w_n)$ using this expression, we need to look at the whole history of words. The N-gram model suggests that we can approximate this probability using the Markov assumption by looking at only N-1 past words. For this study we have picked N to be 3, which is common in the literature. Under the Markov assumption,

$$P(w_n| w_1, w_2,...,w_{n-1}) \approx P(w_n| w_{n-2}, w_{n-1})$$

for a trigram model (N=3). These probabilities are then calculated by the counts of occurrences of word sequences, using the maximum likelihood estimation [20].

To study whether a common, single N-gram model can faithfully represent two samples of language use, we can compare the frequency distribution of N-grams from the two samples. If the distributions are similar, we may conclude that a single language model can reasonably model these two samples. If not, however, then we have evidence that the model of one sample is a poor representation of the other. We use the **Kullback-Liebler divergence** (KL-Divergence) between the two probability distributions as a measure of their similarity. Given two probability distributions P(W) and Q(W) over the same

set of N-grams, we can calculate the difference between these two distributions. KL-Divergence between two probability distributions P and Q, is given by:

$$D_{\mathrm{KL}}(P\|Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}$$

Because this measure is not symmetric, meaning $D_{\mathrm{KL}}(P\|Q) \neq D_{\mathrm{KL}}(Q\|P)$, we can use the symmetric version $D_{\mathrm{KL}}(P\|Q) + D_{\mathrm{KL}}(Q\|P)$ in order to make this measure into a distance metric (distance metrics must be symmetric). KL-Divergence is also closely related to entropy, H(P) which appears commonly in the form of perplexity $2^{\mathrm{H(P)}}$, a common metric in Natural Language Processing (refer to Section 4.4 for more details).

By calculating the KL-Divergence between the two N-gram models provided by the dictated doctor's notes (refer to Chapter 3 for details of the data) and the conversation gold standard, we will be able to show the difference between these two styles of speech.

4.2.2 Naïve Attempt in Calculating KL-Divergence

Our first attempt in calculating the KL-Divergence between the dictation and conversation corpora starts with building a Python program to extract the N-gram model given a corpus.

Given two distinct N-gram language models, the sets of events of the N-grams are different. For example, a word triple "paint the house" might exist in one distribution, but not the other. This introduces a zero division or log(0) error in our calculations of KL-Divergence if a particular word sequence is missing from the other language model. Therefore, we must adjust both of the language models in order to accommodate all possible word sequences that both language models cover.

Adjusting an N-gram language model to accommodate non-occurring word sequences is called **smoothing.** For the naïve implementation, we used the simplest smoothing algorithm, named "add-one smoothing", which adds a count of 1 to non-occurring word sequences while calculating the N-gram probabilities.

Add-one smoothing is highly discouraged, because it gives a very uninformed decision about a novel event, and weighs all novel events equally. Therefore, we need a more complicated method for smoothing. SRI Language Modeling Toolkit (SRILM) offers a variety of smoothing algorithms along with tools for building N-gram models, and is used as a standard tool for many other applications in the field of NLP [7] [8] [9].

4.2.3 Using SRILM

In order to cover all possible word groups existing in both of the language models, these two models have to have a common vocabulary. Using the "ngram-count" executable provided by the SRILM toolkit with the "-write-vocab" option allows us to create a closed set of vocabulary for a corpus. For the other corpus, we execute the same command and combine the two vocabularies so that there is a closed set of words.

For smoothing, we use the original Kneser-Ney algorithm, which yields low perplexity values, as suggested by multiple studies [10] [11]. The Kneser-Ney smoothing algorithm uses the back-off principle in which given a novel event, the algorithm backs off to probabilities of lower order N-grams. For example, if we want to estimate the probability of seeing a word pair XY in a corpus in which it does not actually occur, we can use the individual word probabilities and assume that P(XY) = P(X)P(Y). Although this is a crude approximation to the actual probability that we might expect to see in a much larger corpus, it does avoid the problem of zeros in simple N-gram counts and plausibly approximates the right probability assuming that each word choice is independent of the others. For an open vocabulary language model, i.e. a language model that will accept out-of-vocabulary words, the algorithm assigns a low probability to the token, <unk> and converts all unknown words into this token to avoid zero probabilities.

Using Kneser-Ney smoothing, we have generated trigram models of the dictation and the conversation corpora. Kneser-Ney smoothing can be applied by using the "ngram-count" executable with the "-ukndiscount" option.

Using the set of all possible trigrams coming from both probability distributions, we can measure the probability of a given trigram on each of the language models using the "-ppl" option of the "ngram" executable provided by the SRILM toolkit. Finally, we end up with two probability distributions over all possible trigrams from both of the language models. Figure 3 explains the complete process of building probability distributions, graphically.

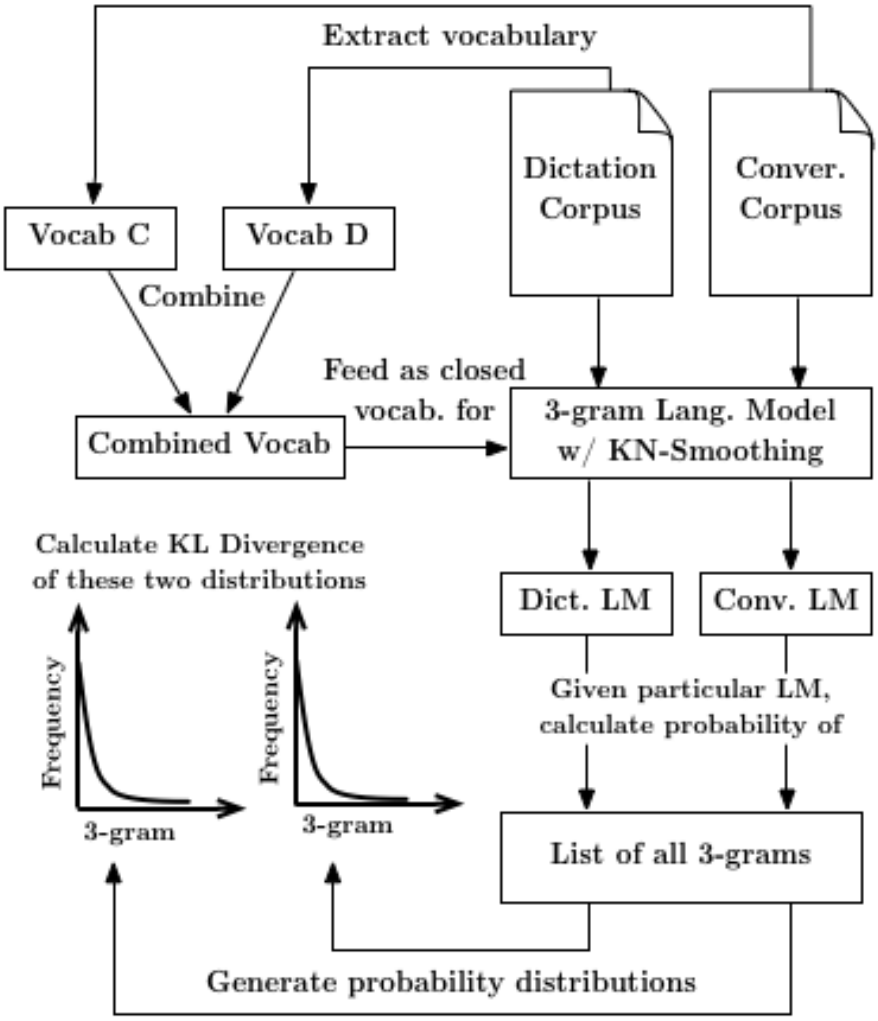

Figure 3: Calculating the KL-Divergence value of two probability distributions derived from two corpora in plain text format.

After obtaining these two probability distributions we can measure how different they are by normalizing the probabilities so that they sum up to 1 and then using the equation for KL-Divergence. Our measurements show that the symmetric KL-Divergence between the dictation and the conversation corpus over the set of all possible trigrams is 7. However, a single scalar value does not tell us how much of a difference that is. Is this a small or a large difference?

4.2.4 Moving from a Single Value to Meaning

Now that we have a single scalar value for the difference between the two language models, we need to see how this value is placed in the spectrum of different KL-Divergence scores. Our claim is that if two documents are being generated from the same language model, they should have a low KL-Divergence. We have no good *a priori* model of the magnitude of KL-Divergence values for different corpora that are in fact drawn from a larger corpus that we believe should be generated by a common language model; i.e. if we take a large set of dictation transcriptions of PEHC notes, we do not know ahead of time how much their word N-gram probabilities differ.

Let D be the dictation and C be the conversation corpus, and the individual documents, corresponding to each visit in those collections be D_i and C_i. There are several possibilities of comparing the two language models derived from these corpora:

1. Comparing each document pairwise from D and C, and also pairwise within the documents in D and in C. i.e., if KL is the KL-divergence between two documents, we would get
    a) the set of scores $KL(D_i,D_j)$ (of which there should be $n*(n-1)/2$ if n is the number of documents in D

b) the set of scores $KL(C_i,C_j)$, of which there are $m*(m-1)/2$ if m is the number of documents in C

c) the set of scores $KL(D_i,C_j)$ of which there are $m*n$.

We could then study the three distributions and their relationships. If (a) differs significantly from (c), then we have good evidence that they are drawn from different models. This can also be computed between (b) and (c), and should give similar results. If (a) differs significantly from (b), that tells us that the one with the wider distribution is more heterogeneous than the other. We expect that C will be more varied than D, and we will explain this phenomenon further in Section 4.4, using the perplexity analysis.

2. For the comparisons (a), we can replace one of the documents by an aggregate of documents, excluding the particular document we are calculating the KL divergence against the rest, by the leave-one-out jackknifing method i.e., rather than computing $KL(D_i,D_i)$, we compute $KL(D_i,D\text{-}D_i)$, where $D\text{-}D_i$ is the language model consisting of all the documents in D except for D_i. Then, we can calculate $KL(C_i,D)$ to see how these two distributions overlap. If C_i's are being generated by the same underlying distribution as D_i's (which we assume to be generated by D),

we will see an overlap between $KL(D_i,D\text{-}D_i)$ and $KL(C_i,D)$. We expect similar results from the symmetric operation.

3. We predicted that option 1 and 2 will both take a lot of effort, because we will have to rearrange the corpora so that each document have a counter-part in the other corpus. Generating language models from $D\text{-}D_i$'s are also computationally expensive. Instead of calculating $KL(D_i,D\text{-}D_i)$ using the jackknifing method, we can generate multiple random corpora, called G_i, based on the language model derived from D, using the "-gen" option for the "ngram" executable in the SRILM toolkit. This procedure is called bootstrapping and it is an approximation to the leave-one-out jackknifing method [22]. Using these randomly generated corpora, G_i we can create the distribution $KL(G_i,D)$, which is an approximation of $KL(D_i,D\text{-}D_i)$, and see how much it overlaps with $KL(C_i,D)$.

In Section 4.2.3, we calculated that the single value KL(C,D) is approximately 7. Due to time restrictions, we were only able to use option 3, with the adjustment of using the single value KL(C,D) as an approximation, instead of the distribution $KL(C_i,D)$. It would be worthwhile to see what kind of results we

gather from option 1 and 2, as they better reflect the nature of the differences of the two language models.

We propose this novel method as a way of measuring the difference between two language models quantitatively. Figure 4 explains the process we have done for this study, graphically.

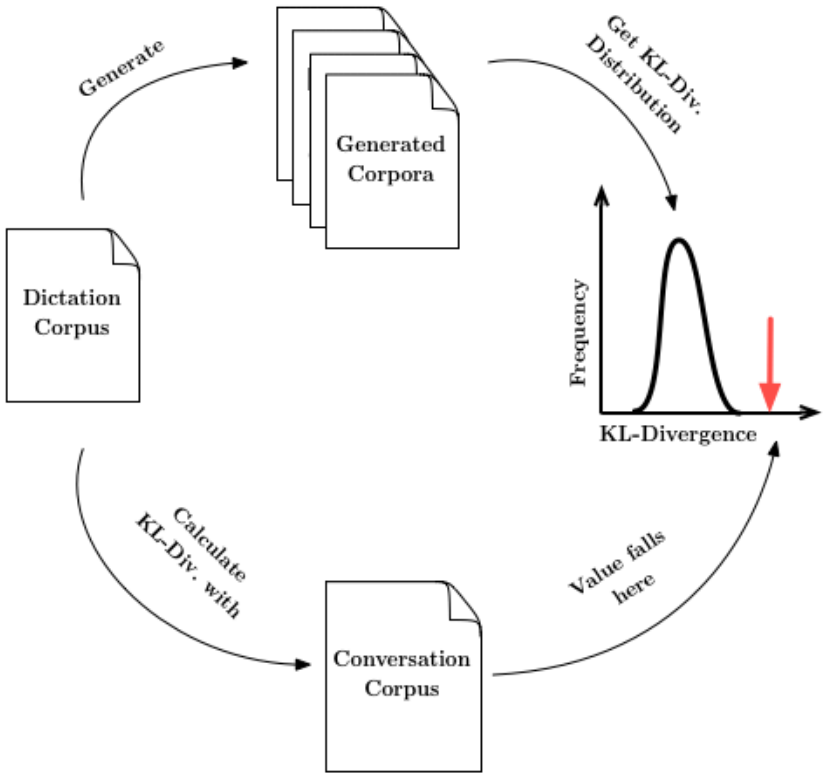

Figure 4: The novel approach for comparing two language models by KL-Divergence values.

Our calculations showed that the KL-divergence scores between the dictation corpus and the generated corpora from that corpus, $KL(G_i,D)$ varied between 0.6 and 0.8 for four generated documents, whereas the single value we

obtained from dictation versus conversation, KL(C,D), was 7, lying outside this range by far.

This experiment suggests that the language models of dictational and conversational speech are different. Therefore, an ASR system designed to handle dictational speech yields low accuracy trying to transcribe conversational speech, due to this difference between two types of speech. This is obviously not the only reason for low performance of ASR systems, but it is a major one that should be considered.

4.3 ASR Language Model Optimization

Many ASR systems allow the user to adapt the default language model to the specific domain and style of speech that is to be expected. For example, Nuance actually sells a version of DNS that is specifically adapted to the needs of doctors doing dictation. That version is based on a large library of previously transcribed dictations, and produces a model that more accurately reflects what DNS expects to hear from a doctor as opposed to, say, a businessman. Both systems also allow the user to provide examples of text in their domain, from which they further adapt their language models. However, in general the amount of domain-specific text available to the user (including us) is much smaller than

the amount of data on which the models were initially trained, so the adaptation effects of doing this additional language training may be minimal.

Nevertheless, we explored this capability to investigate whether we could improve the accuracy of both systems on our conversation corpus.

4.3.1 SRI DynaSpeak Language Model Optimization

SRI DynaSpeak allows the modification of its language model to adapt to the needs of a specific task. Therefore, we can use this opportunity to train DynaSpeak with the gold standard conversation corpus that we have and thereby to make improvements on the baseline values that the default model yielded, presented in Section 4.1.2. In this experiment, we try to use a combination of tools that DynaSpeak and SRILM provides, and adapt the language model to handle a conversational speech model.

The version of SRI DynaSpeak for this experiment is the 04.02.2010 release, and is older compared to the default general model. The reason we are using an older model is that this release is the newest model that we can modify. Due to licensing restrictions, we are only able to adjust the language model, leaving the acoustic model unchanged. The details of how the language model and the acoustic model are used by an ASR are explained in detail in Section 4.2.1.

SRI DynaSpeak takes in two types of grammars: Java Speech Grammar Format (JSGF) or Probabilistic Finite-State Grammars (PFSG). A JSGF grammar is good for smaller tasks that require a small set of vocabulary and grammar rules, such as a car navigation system where the input speech could be predicted beforehand. A PFSG can be used for much larger tasks for which the valid inputs are not easily predicted with a set of rules. PFSGs are created using language models that are created by SRILM in the standard ARPA format (**A**dvanced **R**esearch **P**roject **A**gency), developed by Doug Paul at the MIT Lincoln Labs. PFSGs are an equivalent data structure to the ARPA format, in the sense that they are the finite state machine representation of a probability distribution.

Using the PFSG, we have transcribed 6 conversation files and obtained WERs that are significantly higher than the baseline model by far, therefore we are not displaying the results here. Our conversations with SRI suggested that the gold standard corpus is too small and has very high perplexity to be used as a functional language model. Previous studies show that perplexity and WER are highly correlated [14]. In their research, Klakow and Peters show that 450 distinct documents with different perplexities, which are about a single topic, exhibit the behavior in Figure 5.

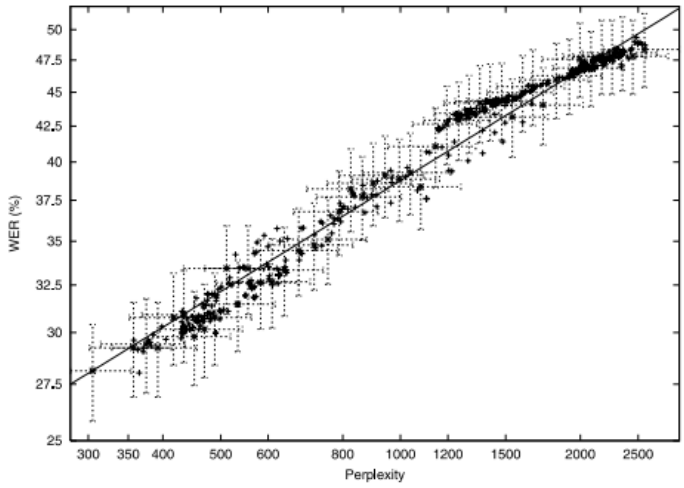

Figure 5: WER versus perplexity graph of 450 language models. As perplexity increases, WER increases. From of Klakow and Peters, 2001.

We analyze the perplexities of the dictational and the conversational corpora in detail, in Section 4.4.

4.3.2 DNS '11 Language Model Optimization

In this experiment, we have used two of the tools that DNS '11 offers in order to improve the accuracy of speech recognition.

The first one of those tools, voctool, takes a collection of written documents and updates its current language model to account for these documents. Since we have a collection of gold standard transcriptions, we can feed these into voctool so that DNS '11 can use this conversational language model to transcribe spontaneous speech. Let T be the collection of gold standard

transcriptions, and A be the corresponding audio file for a given conversation. To transcribe a given A_i, we take out that T_i from T, and feed the T-T_i gold standard transcriptions into voctool. This way we do not reveal the correct transcription to DNS '11 but we use ideally similar conversations to transcribe a new conversation. Due to time restrictions, we were able to calculate the WERs for 4 conversation files, using a leave-two-out approach. Figure 6 shows the WERs by DNS '11 default model, and after the language model optimization by voctool.

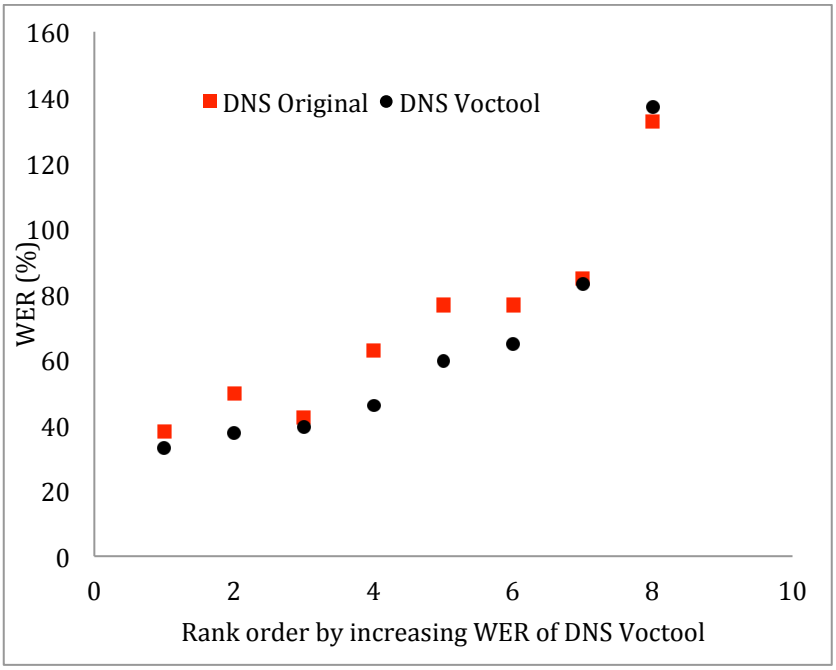

Figure 6: The WERs of 4 conversation files (two data points for each side of the conversation) after language model adaptation using voctool.

The second optimization tool, Eyes Free Enrollment (efenroll) allows us to optimize the acoustic model by doing a multi-pass transcription of a conversation. After the first run using the default language model, the output transcription and the audio file can be fed into efenroll as a pair, so that efenroll reinforces the acoustic model by adjusting some parameters under the hood. Next, efenroll creates a modified user for which DNS '11 has a better acoustic model. This is analogous to training DNS '11 beforehand with a predetermined text. The only difference is that efenroll enables the usage of a custom "training text" (which is the first automatic transcription given by the unmodified default model) and later uses this updated model to transcribe the conversation for the second time. The initial transcription might not be so accurate, in which case this technique would not work well.

Our attempts to transcribe four random conversations using efenroll showed us that there is in fact an increase in WERs, so we did not continue to transcribe more conversations. Some explanations of why this method performs poorly are presented in Section 5.3.2.

4.4 Perplexity of the Corpora

In this experiment, we analyze the perplexities of the dictational and the conversational corpora. We have shown previously that WER increases in

correlated with high perplexity [14] [15]. Therefore, if conversational speech has a higher perplexity than dictational speech, it would be reasonable to say that one of the reasons that the ASRs are performing poorly on conversational speech is because it is trained to handle dictational speech, which has a lower perplexity. The mathematical definition of perplexity is

$$2^{H(p)} = 2^{-\sum_x p(x) \log_2 p(x)}$$

where p(x) is the probability distribution that is the same as the N-gram language model, $P(w_1, w_2,...,w_n)$ and x is the set of all trigrams in the language model, in our case.

Conceptually, perplexity tells us how confused a language model is. Perplexity per word, $2^{H(p)/N}$ is the average number of different words that the model has to choose from, given a test sequence. For example if the perplexity per word is 8, there are 8 possibilities that could qualify as w_3 in order to become the next word in the sequence, w_1, w_2, ... [20].

Instead of coming up with a single perplexity value, which is calculated by measuring how perplexed a language model is in itself, we calculated a distribution of perplexities with a jackknifing method, where we (1) take out 50 sentences out of the corpus, (2) build a language model with the remaining corpus and (3) calculate the perplexity of 50 sentences given as the test set, over

all trigrams forming the test set. This approach is very similar to the method proposed in option 1 in Section 4.2.4, comparing (a) and (b) with KL-divergence. With this method, we propose that we can see the different spreads of these language models and decide which one is more heterogeneous. What we have done differently for this experiment is that, instead of KL-divergence, we used perplexity, which is a more common measure while deciding the variability within a language model. Figure 7 outlines this procedure as a diagram:

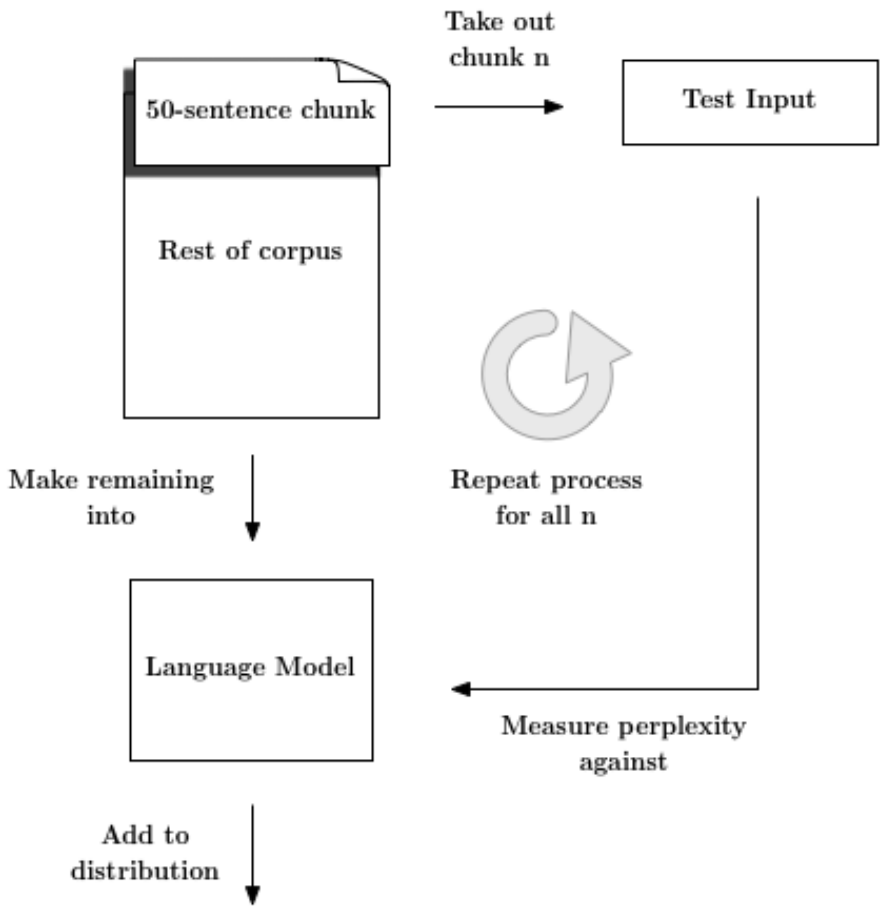

Figure 7: Calculating the perplexity distributions of the two corpora using the jackknifing method

While taking out the 50-sentence chunks, we have implemented a Python program that does that in memory, utilizing the Python built-in mmap module and saving us a great deal of time. Normally, Python opens the whole document and after taking out the 50-sentence chunk, writes the whole document back to the hard drive. This operation takes a long time. With mmap (short for memory map), we map the file to the memory and make changes on it, and we can use this temporary document as a corpus for the language model without having to write it back constantly. Finally, we came up with two distributions of perplexities for each of the dictation and the conversation corpora. Figure 8 shows these two distributions.

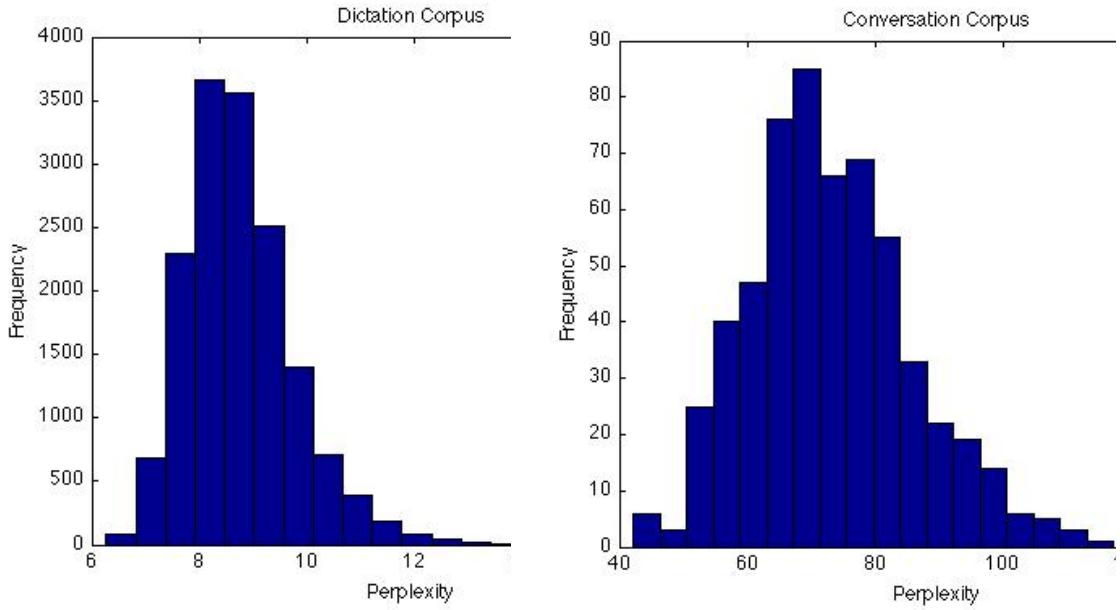

Figure 8: Perplexity distributions of 50-sentence chunks against the two language models. The much greater perplexity of the conversation corpus is evident in its far larger perplexity scale.

The mean perplexity of the dictation corpus is 8.8, whereas it is 73.1 for the conversation corpus and the distributions are significantly different. These results match our expectations because we predicted that conversational speech is less structured due to its spontaneous production, compared to dictational doctor notes, which have to follow a certain format. Knowing the structure of a few dictations can tell a lot about a new dictation, but this does not hold for conversations. A statistical significance test reveals that there is no overlap between the two distributions of perplexities (p-value ≈ 0).

4.5 Optimizing Audio Input

Due to the nature of the recording setup in the clinic, there is a lot of crosstalk between the microphones. This is a major contribution for the "insertion" type of errors introduced by the WER results in Table 1 and 2. By definition, "insertion" errors occur when the hypothesized text inserts words that are not in the reference text. These types of errors happen when ASRs try to transcribe breathing and other background noise as possible words. In our case, the ASR also picks up the second party's voice and tries to transcribe it as the main voice. In this section, we propose a simple approach that could be used in real time after some adjustments in order to eliminate some of the crosstalk.

For each of the speakers, the microphone is attached as a headset, which means that the main speaker for that microphone is very loud compared to the rest of the audio signal. With a noise gate that opens when the amplitude is higher than a certain threshold level, we can eliminate some of the crosstalk. Although this is not a high accuracy speaker identification and segmentation technique, it is a very efficient and fast way to get rid of the background noise on the go.

Next, we need to make sure that we pick a value for the threshold amplitude so that we eliminate most of the crosstalk and do not delete the main speaker accidentally. Picking a constant value for the threshold amplitude is not good, because, for example, for a new visit, we could have a louder second party or the doctor might be quieter than during the previous visit.

Therefore, we need to design a program that adapts to these changes. For the simplest approach for classifying an audio segment as low or high in amplitude, the single feature is the amplitude. More complicated ways of identifying a speaker are possible by extracting many features provided by the frequency content. However, these techniques take a lot of computational time, and considering that Fairwitness is designed to work in real time, we take the faster approach.

We assume that the loudness of each speaker can be modeled with a Gaussian distribution. When we analyze the amplitude distribution for the raw recording, the loudest Gaussian most likely corresponds to the main speaker because he or she is the closest to the headset. Then, we keep the loudest Gaussian, and silence the rest. This method is a crude approximation to a Gaussian Mixture Model in one dimension. Figure 9 shows an example of an amplitude distribution calculated by an averaging window of 160 milliseconds.

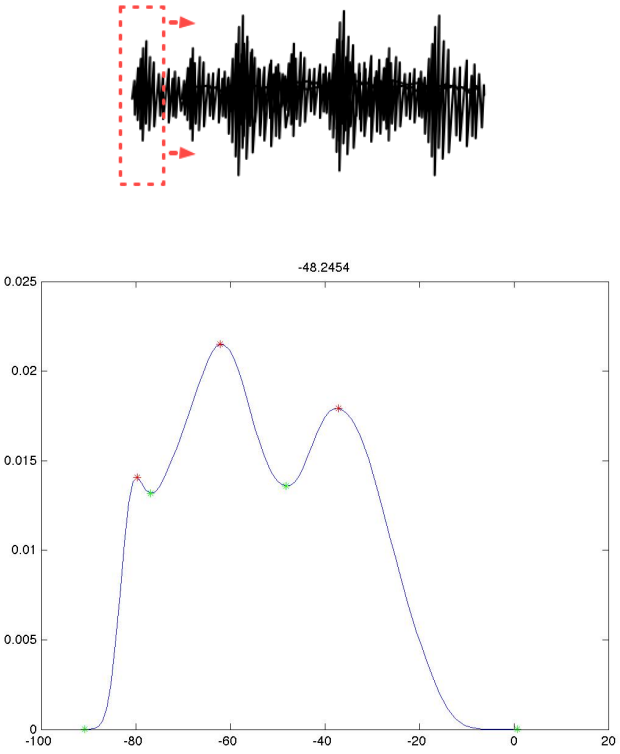

Figure 9: Averaging window of 160 ms for calculating the amplitude distribution for a single audio file. The histogram is smoothed so that the Gaussian distributions are visible. In the graph, x-axis is the amplitude and the y-axis is how often that particular amplitude occurs. From loudest to softest, the three peaks roughly correspond to the main speaker, the second speaker and the silence parts in a conversation.

After we obtain the histogram for the distribution of the amplitude values, we smooth the histogram and find the loudest local minimum. This local minimum is the intersection of the loudest Gaussian with the second loudest one. We then silence the signals below this threshold value and keep the ones above, ideally letting the main speaker be the only signal. With the given cutoff value, we can run SoX with the "compand" command to let the audio signal pass if the amplitude is greater than that particular threshold. "compand" is a noise-gate with an averaging window as described by attack and decay, similar to the "silence" functionality described in Section 4.1.2. Notice that this algorithm can be modified slightly to function "online", meaning that it will noise-gate an audio signal on the fly. The online version of the algorithm builds up the amplitude distribution as the conversation progresses, and dynamically finds the best cutoff point.

After we obtain the modified audio signals, we feed these into the default models of DNS '11 and SRI DynaSpeak. Table 3 displays the WERs obtained from nine modified audio files and transcribed by DNS '11 default model:

Document	Correct (%)	Substitution (%)	Deletion (%)	Insertion (%)	WER
100329.p	19.2	80	0.8	171.1	251.9
090624.p	26.7	69.2	4.1	53.1	126.3
090624.d	16.3	82	1.7	75.1	158.8
090518.d	48.2	33.2	18.6	6.1	57.9
090902.p	49.3	37.6	13.1	20.3	71
090406.d	43.5	33.9	22.6	2.8	59.3
100407.d	65.9	20.3	13.8	8.4	42.5
090921.d	64.3	22.5	13.2	5.7	41.5
100303.d	66.1	21.1	12.9	5.1	39

Table 3: The WER statistics of the modified audio input by DNS '11

Table 4 displays the WER statistics of the same input using SRI DynaSpeak.

Document	Correct (%)	Substitution (%)	Deletion (%)	Insertion (%)	WER
100329.p	14.7	84.9	0.4	137.7	223
090624.p	21.8	62.9	15.3	21.6	99.8
090624.d	16.2	82.5	1.3	90	173.8
090518.d	6.2	13.2	80.6	0	93.8
090902.p	15.3	25.9	58.8	1	85.7
090406.d	12.2	33.8	54	0.6	88.4
100407.d	32.7	37	30.3	5.2	72.5
090921.d	45.2	32	22.8	3.8	58.5
100303.d	41	35.4	23.6	4.4	63.3

Table 4: The WER statistics of the modified audio input by SRI DynaSpeak

Figure 10 helps to visualize the potential improvement on the WERs, after the audio optimization treatment. On the left, we see the DNS original versus DNS with the cutoff algorithm, and SRI DynaSpeak on the right.

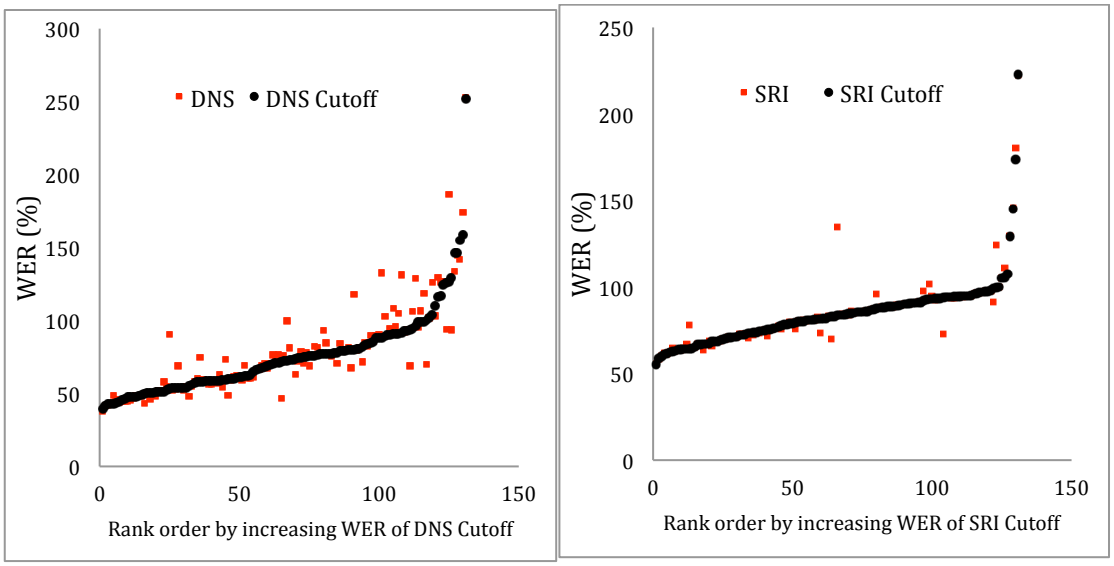

Figure 10: The scatter plots of WERs for each ASR before and after the cross talk elimination algorithm. The data points correspond to 66 transcribed conversations (two data points for each side of the conversation).

Chapter 5

Discussion

In this chapter, the results of the improvement and optimization tasks of this study will be discussed in light of the data collected and statistical tests. The discussions focus on the results we obtained from the experiments: ASR default model performances, language model differences, SRI DynaSpeak optimization, DNS '11 optimization, corpora perplexities and audio optimization. Depending on the type of data or the process, either paired (treatment on a sample) or two-sample unequal variance (different samples) t-tests were carried out.

5.1 ASR Default Models

This section includes evaluations on basic pre-optimization aspects of this study, such as microphone quality, ASR quality and differences between doctor and patient speech and language use patterns.

5.1.1 DNS '11 vs. SRI Default Model Comparison

One prominent aspect of this study is to ascertain whether there are any significant differences between DNS '11 or SRI Dynaspeak, and if so, which ASR is better at speech recognition in our application with their default model. Based on 66 doctor and 65 patient conversation documents (one file was excluded, because it was an outlier), mean WER for DNS '11 is lower than it is for SRI. This difference is mainly due to SRI performing quite poorly in the deletion category. Table 5 shows that even though DynaSpeak is doing much better in substitution and insertion, deletion makes SRI DynaSpeak perform significantly worse.

ASR	Substitution (mean)	Deletion (mean)	Insertion (mean)	WER (mean)
DNS '11	40.2	18.6	18.8	77.6
SRI	36.0	40.3	8.7	85.0

Table 5: Mean WER for default DNS '11 and SRI DynaSpeak

Since the transcriptions by both ASRs can be considered as treatments on the same sample, a two-tailed paired t-test was used to measure if these differences were statistically significant. The results were affirmative ($p<0.01$ for substitution, deletion, insertion and WER).

These results might indicate that DNS '11 has performed better overall compared to SRI DynaSpeak at first glance, yet further analysis might reveal that to be not the case. The higher overall WER for SRI DynaSpeak stems from

deletion. As previously explained, SRI DynaSpeak requires small chunks of speech to be loaded for analysis, so the process included slicing the conversations into chunks. It is quite likely that the slicing has resulted in some words being reduced to incomprehensible parts and that SRI DynaSpeak has omitted those. The significantly better performance of SRI DynaSpeak in the substitution and insertion categories should be taken into consideration in further analysis.

5.1.2 Microphone Performance

In order to improve recording quality, the microphones were switched from Andrea NC-91 to Sennheiser ME3s on the recommendation of audio engineers who believed that the increased directional sensitivity of these newer microphones would improve rejection of the other side of each conversation and therefore help with correct interpretation of our conversations. To analyze whether this has led to significant improvements, the WERs of 51 samples recorded with Andrea NC-91 were compared to that of the 80 samples recorded with Sennheiser ME3s. A one-tailed two-sample unequal variance t-test has indicated that for neither DNS '11 ($p=0.08$) nor SRI DynaSpeak ($p=0.49$), can we reject the null hypothesis that the two microphones perform equally well at $a=0.05$. However, for DNS there is only a 8% chance that these results were obtained at random, which still provides some confidence that the new microphones helped.

5.1.3 Doctor vs. Patient Speech

In order to optimize the results for speech recognition, it is also necessary to find out whether there are significant differences between WER for doctor and patient speech. As Table 6 shows, the mean error rate for patients is much higher than it is for doctors for both ASRs. This would have been an expected difference if trained voice models were used for the doctors. However, neither the doctors nor the patients had trained the system beforehand. A two-tailed unequal variance t-test shows that the difference is significant, meaning that both ASRs perform better with doctor's speech.

ASR	Doctor (WER)	Patient (WER)
DNS '11	64.5	91.0
SRI	79.2	91.0

Table 6: The WERs of doctor and patient speech for each ASR

There might be a number of reasons for such results. The doctors might speak closer to the microphone, because they are more experienced in such settings. The doctors might prefer a more concise and dictational tone. One piece of information that might be useful is that the rate of insertion for patients under DNS '11 (mean 26.287) is significantly higher than it is for doctors (mean 11.447, with a p-value < 0.01). Since the doctors speak more than patients do (1997 words vs. 1243 words on average, also statistically significant), it is quite possible

that the patient's microphone captures words spoken by the doctor, which leads to a much higher insertion rate for the patients.

5.2 Language Model Difference

In this study, we introduce a novel approach to comparing two language models and hypothesize that conversational and dictational models have significant differences. Using the method previously described in Section 4.2.4, the KL-divergence values between the language models of the generated documents and the dictation corpus are 0.7, 0.7, 0.8 and 0.8, whereas the KL-divergence between the conversational language model versus the dictational language model is 7. Therefore, there is a significant difference between the two language models.

This approach can be considered as an alternative to currently used statistics such as term frequency-inverse document frequency (tf-idf), cosine document similarity or vocabulary overlap, yet the validity of the model should be further evaluated, such as by testing on a variety of documents that are believed to originate from similar language models.

5.3 ASR Language Model Optimization

5.3.1 SRI DynaSpeak Language Model Optimization

As seen in the results before, the SRI Language Model requires optimization to perform better for the task at hand. We have consulted SRI on their opinions for this issue, for which they have proposed working with a larger dataset as a possible future step.

5.3.2 DNS Language Model Optimization

We have seen that updating the default language model using voctool provided an average of 8% decrease in WER in four 4 conversation files. This improvement is statistically significant ($p<0.01$). It would be worthwhile to see if the average improvement changes if we are to transcribe all the additional conversations we have at hand.

Efenroll did not provide any improvements in the accuracy of DNS '11. One of the reasons for this might be the fact that we are trying to use an already not-so-good initial transcription to be the basis of acoustical model optimization. We predict that efenroll would perform better in scenarios where the initial WERs were already low.

5.4 Corpora Perplexity

One of the goals of this study was to quantify the differences between conversational and dictational models as discussed in our KL-divergence based approach. Another step that stems from this approach is to work on the perplexity of the corpora. As suggested by [14] [15], there is a positive correlation between WER and perplexity, and that theory fits in our corpora as well. As can be seen in the perplexity histograms in Figure 8 in Section 4.4, there is a significant difference between conversational and dictational corpora: the former has a mean of 73.1 compared to a mean of 8.8 in the latter. That is one possible explanation as to why our rate of accuracy is lower in conversations. A language model derived from a dictation corpus is capable of guessing new similar input, whereas that which is derived from a conversation corpus is not very successful in doing so. To elaborate, when we try to predict a chunk of conversational speech given the conversational language model, a high rate of entropy prevents a successful attempt. This is not so much the case for the dictational corpus.

On the other hand, the results above could be biased due to the length of the corpora. The dictation corpus has 750,000 sentences whereas the conversation corpus has 30,000. A smaller corpus might naturally have a higher perplexity, but we have not considered that for this experiment. One study by Ney et al. shows that the perplexity of a language model, which was derived from the Wall Street

Journal corpus, decreases as the number of words in the corpus increases from 1 to 38 million [24]. Therefore, it would be worthwhile to compare two corpora that are equal sizes to control for the length, while measuring perplexity.

5.5 Audio Optimization

One proposed solution to decrease the high error rates observed in both DNS and SRI data was to eliminate the background noise and crosstalk observed in the audio. A comparison of the mean WERs before and after the application of the crosstalk cutoff technique can be seen in Table 7:

ASR	Substitution (mean)	Deletion (mean)	Insertion (mean)	WER (mean)
DNS Default	40.2	18.6	18.8	77.6
DNS Cutoff	37.2	24.6	13.5	75.3
SRI Default	36.0	40.3	8.7	85.0
SRI Cutoff	34.0	43.7	6.8	84.5

Table 7: WER statistics for DNS and SRI with audio cutoff algorithm

The cutoff technique has overall led to a decrease in the mean error rate for both ASRs, yet a one-tailed paired t-test suggests that even though it is possible to reject the hypothesis that the cutoff technique has not worked for DNS, it is not statistically significant to reject the hypothesis for SRI. In any case, the net effect on WER for both systems was modest.

In order to understand why that is the case, a comparison of each evaluation category within the ASRs could be useful. The t-tests revealed that

there were significant changes in all categories for both systems; that is, the cutoff technique has resulted in an improvement for substitution and insertion, yet it also resulted in a higher rate of deletion. For DNS, higher accuracy in substitutions and insertions could offset the lower accuracy rate in deletions, but that was not the case for SRI. This shows that even though the audio optimization algorithm reduced the insertions, it also introduced more deletions of the actual speech while trying to minimize the cross talk.

Two possible solutions to this issue could be improvements in the slicing technique for SRI as discussed before, or improvements in the algorithm for the cutoff technique. These will be discussed further in the future work section.

Chapter 6

Future Work

In this chapter, we propose future improvements to the current workflow in light of the findings we presented in Chapter 5. These improvements are worthwhile to investigate because they will help us achieve more accurate results in our efforts to transcribe clinical conversations into text.

6.1 Improving Audio Input Further

In this study, we have implemented a simple and fast algorithm to eliminate cross talk in two microphone recordings. We used the amplitude as the only feature while separating out the main speaker, because we wanted to implement this algorithm in real time. It would be worthwhile to see other existing methods to segment multi-speaker recordings. Speaker diarization is the name given to the task dealing with the segmentation of multiple-speaker recordings. The state-of-the-art speaker diarization systems use multi-dimensional features to identify speakers. These features are based on the frequency domain of the audio signal, and can reveal more information about the characteristics of a particular speaker, which makes it easier to distinguish one speaker from another.

Adding more features to a classification makes the task computationally more expensive. Metze et al. report that their speaker diarization system decreases the WER for non-segmented speech from 66% to around 45% [25]. It should be worthwhile to investigate speaker diarization methods and use one that is fast enough to be used in real time in our use case.

6.2 Decreasing the WER for SRI DynaSpeak

We have seen in Chapter 5 that SRI DynaSpeak default model is performing significantly lower than DNS '11. We showed that the dominant part of this error is introduced by the deletions. The slicing algorithm presented in Section 4.1.2 is a candidate for contributing to deletion errors. SRI DynaSpeak requires short audio files as inputs, which required the slicing. During the slicing procedure, SoX deletes most of the silence between two utterances, where silence is defined as 2% of the loudest signal. This increases the likelihood of deletion of important utterances by accident. Therefore, we could devise a smarter algorithm for slicing audio files to minimize the number of deleted utterances, and then repeat the SRI DynaSpeak results for more accurate results. There have been many attempts in identifying human voice in audio streams. This procedure is called Voice Activity Detection (VAD), and has been successful in various

applications [12] [13]. We could incorporate this algorithm to slice the conversation files before feeding them into SRI DynaSpeak.

Our conversations with SRI led us to the conclusion that the reason for the low performance of the modified language model is the inadequate size of our conversation corpus that was used to train their language model. Therefore, we could record and hand transcribe more conversations in order to build a more robust language model that SRI DynaSpeak could rely on while transcribing clinical conversation.

6.3 Using other ASRs

Recently, several leading technology companies such as Google and AT&T made available their APIs for speech-to-text, publicly. These systems are based on cloud computing, where the audio data is processed in remote servers, and have been very successful. The fact that these companies might store any input speech files for improving their own systems prevents us from using those services for processing our confidential medical data. It would be worthwhile to investigate the possibility of de-identifying audio files to permit use of these online ASRs. One way of doing this would be aligning the gold standard text with the audio file, and then using a text de-identifier tool, such as Arya Tafvizi's [21], developed here in the Medical Decision Group, to first de-identify the aligned text and then cut out the part of the speech signal that corresponds to

the identifying text. To be able to do this successfully, we would need to make sure that the gold standard text is in fact transcribed very accurately.

6.4 Impact of Transcription on Semantics

WER is a metric that relies heavily on the exact matching of words in a best-aligned sequence. That means that small word errors reflect negatively on the performance of the system, even though the semantics might stay intact. In order to measure how the transcription affects the semantics, we need a quantitative measure of the semantic content. For the next steps of this project, we could use Latent Dirichlet Allocation (LDA) as a way of measuring semantic content. LDA is widely accepted as a topic-modeling tool in the research community [5] [16] [17]. LDA interprets documents as an arbitrary number of topics with associated keywords and weights. In order to see the semantic coverage of an ASR, we can compare the LDA analysis of the gold standard with that of the ASR transcription. The simplest way of comparing two topic models would be by representing the topics as vectors, and calculating the dot product of these two vectors. This will tell us how much of the semantics we lost with the ASR transcription. There are plenty of problems associated with this method, such as dealing with zeros, because they do not contribute to the dot product even though they are clearly mismatches. There might also be other problems arising, but it is an interesting idea that could be investigated.

Chapter 7

Contributions

In this thesis, we have:

1. Proposed the Fairwitness Project, which (a) listens to a conversation between a patient and a doctor, (b) uses the ASR technology to transcribe that conversation into text, (c) applies natural language processing methods to extract the important clinical facts from the conversation and (d) organizes and presents this information in real time to the participants.
2. Conducted 5 different experiments to investigate the drawbacks of the current ASR systems, such as the fact that those ASRs are trained only on dictational language models and they are optimized for single speaker settings.
3. Introduced a new method for quantitatively measuring the difference between two language models using KL-divergence.

4. Measured the difference of dictational and conversational language models using a bootstrap version of this method, and showed that these two types of speech are indeed different in the clinical domain.
5. Improved DNS '11 accuracy by adapting conversational speech to its language model, using the provided optimization utility, voctool.
6. Measured the perplexity of clinical conversations and dictations and showed that the conversational language model has a significantly higher perplexity than dictational one.
7. Predicted that the high WERs are due to the highly perplexed nature of spontaneous speech, based on the fact that there is a strong correlation between WER and perplexity.
8. Introduced a fast and simple algorithm for cross talk elimination in two speaker settings, showing significant improvement in WERs of DNS '11.

Bibliography

[1] Liu, Tur, Hakkani-Tur, Yu. “Towards spoken clinical-question answering: evaluating and adapting automatic speech-recognition systems for spoken clinical questions.”
[2] Zheng, Mei, Yang, Manion, Balis, Hanauer. “Voice-Dictated versus Typed-in Clinician Notes: Linguistic Properties and the Potential Implications on Natural Language Processing.”
[3] H. M. Tufo and J. J. Speidel. Problems with medical records. Medical Care, 9:509–517, 1971.
[4] F. J. Romm and S. M. Putnam. The validity of the medical record. Medical Care, 19:310–315, 1981.
[5] Blei, David M.; Ng, Andrew Y.; Jordan, Michael I (January 2003). Lafferty, John. ed. "Latent Dirichlet allocation". Journal of Machine Learning Research 3 (4–5): pp. 993–1022.
[6] Institute of Medicine (U.S.). Committee on Improving the Patient Record and Dick, R.S. and Steen, E.B. and Detmer, D.E. “The Computer-Based Patient Record: An Essential Technology for Health Care” National Academy Press 1997.
[7] A. Stolcke (2002), SRILM -- An Extensible Language Modeling Toolkit. Proc. Intl. Conf. on Spoken Language Processing, vol. 2, pp. 901-904, Denver.
[8] Katsamanis, Athanasios and Black, Matthew. SailAlign: Robust long speech-text alignment. 2011.
[9] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, Evan Herbst. (2007) "Moses: Open Source Toolkit for Statistical Machine Translation". Annual Meeting of the Association for Computational Linguistics (ACL), demonstration session, Prague, Czech Republic, June 2007.
[10] Reinhard Kneser, and Hermann Ney. Improved backing-off for m-gram language modeling. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Vol. I Detroit, Michigan: (May 1995). p. 181-184.

[11] Stanley F. Chena, Joshua Goodmanb, An empirical study of smoothing techniques for language modeling. 1999.
[12] Ramírez, J.; J. M. Górriz, J. C. Segura (2007). "Voice Activity Detection. Fundamentals and Speech Recognition System Robustness". In M. Grimm and K. Kroschel. Robust Speech Recognition and Understanding. pp. 1–22
[13] Freeman, D. K. (May 1989). "The voice activity detector for the Pan-European digital cellular mobile telephone service". Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP-89). 1. pp. 369–372.
[14] Klakow, D; Peters, J. "Testing the correlation of word error rate and perplexity"
[15] Stanley Chen, Douglas Beeferman, Ronald Rosenfeld. "Evaluation Metrics for Language Models"
[16] He Tingting, Li Fang. Semantic Knowledge Acquisition from Blogs with Tag-Topic Model.
[17] Kong, Sheng-Yi, Lee, Lin-Shan. Semantic Analysis and Organization of Spoken Documents Based on Parameters Derived From Latent Topics.
[18] K. Johnson, A. D. Poon, S. Shiffman, R. S. Lin, L. M. Fagan. Q-Med: A Spoken-Language System to Conduct Medical Interviews. 1992.
[19] Zafar A, Overhage JM, McDonald CJ. Continuous speech recognition for clinicians. 1999.
[20] Jurafsky, Dan. Speech and language processing : an introduction to natural language processing, computational linguistics, and speech recognition. Upper Saddle River, N.J: Pearson Prentice Hall, 2009.
[21] Arya Tafvizi. A De-identier For Electronic Medical Records Based On A Heterogeneous Feature Set. MIT Masters of Engineering Thesis. 2011.
[22] Shao, Jun. The jackknife and bootstrap. New York, NY, USA: Springer Verlag, 1995.
[23] Christina Sauper. Content Modeling for Social Media Text. MIT Doctor of Philosophy Thesis. 2012.
[24] Sven Martin, Jörg Liermann, Hermann Ney. Algorithms For Bigram And Trigram Word Clustering. 1995.
[25] Florian Metze, Qin Jin, Christian Fugen, Kornel Laskowski, Yue Pan, and Tanja Schultz. Issues in Meeting Transcription – The ISL Meeting Transcription System. 2004.