SPEECH RECOGNITION FOR WEB BASED TELEPHONY

Mahboob ur Rahman

This thesis is presented as part of Degree of Master of Science in Electrical Engineering

Blekinge Institute of Technology
March 2010
Abstract

Web based telephony purges the need of explicit downloading and installing a VoIP client software. Calls in web based telephony can be made directly from the browser. The combination of web technologies and traditional telephony makes it possible to introduce new exciting services. One such new service is introduced as a result of this thesis work. The voicemails received are automatically transcribed and converted into text; the text is then saved to an inbox. The performance of the introduced service is good and gives a better recognition rate in the current configuration. The speech recognition covers a continuous speech of English and a maximum vocabulary of 64 thousand words. Adobe Flash 10 has a proprietary protocol for the streaming of audio over internet. Red5 server is an open source server that has support for RTMP plug in. Red5Phone is an open source SIP phone containing a flash based client. The new service introduced is added to the existing Red5Phone solution. Speech recognition for web based telephony was investigated, developed, implemented, and tested.

Sphinx-4 is an open source state-of-the art ASR system. It is capable of keeping up with the requirement of large vocabulary transcription. Sphinx-4 was configured and integrated with the developed service for the transcription of voicemails. The performance of Sphinx-4 was rigorously evaluated before its configuration.
Preface

This dissertation summarizes the research work carried out at Ericsson Research, Luleå, as Master Thesis by Mahboob ur Rahman, Electrical Engineering Master student at Blekinge Tekinska Hogskola, Sweden.

Acknowledgment

I am thankful to Daniel Enström for providing me the opportunity to carry out this interesting research work. I would also like to acknowledge the indispensable support and incessant help from my supervisor Jonas Lundberg at Ericsson Research Luleå. His useful guidelines, responsive behavior, timely assistance, and dedication made this work possible. Special thanks to Dr. Jörgan Nordberg for his baking. His worthy comments made the presentation of this report more pleasant. Not forgetting the worthy tips and help from Nicolas at Ericsson Research Luleå.

My utmost appreciation goes to my friends Syed Fakhar Gillani, Yasir Masood Malik, Ayli Raeesi, and Nagarajesh for the helpful and commendable discussions.

Final words of thanks to my friends and family whom encouragement, love and support throughout my master studies at BTH kept me going forward.

Mahboob ur Rahman

Stockholm, March 2010
Table of contents

List of figures................................................................................................................................. 11
List of Tables.................................................................................................................................... 13
List of acronyms.................................................................................................................................. 15

CHAPTER 1 INTRODUCTION........................................................................................................ 17
  1.1 THESIS OBJECTIVE.................................................................................................................. 17
  1.2 PROBLEM DEFINITION............................................................................................................. 17
  1.3 HYPOTHESIS............................................................................................................................. 17
  1.4 THESIS SCOPE.......................................................................................................................... 17
  1.5 THESIS OUTLINE....................................................................................................................... 17

CHAPTER 2 BACKGROUND............................................................................................................. 19
  2.1 THE SPEECH SIGNAL............................................................................................................... 19
  2.2 Sound spectrum.......................................................................................................................... 20
  2.3 Speech production....................................................................................................................... 21
    2.3.1.1 Voiced sounds..................................................................................................................... 22
    2.3.1.2 Formant.............................................................................................................................. 22
    2.3.2 Unvoiced sounds................................................................................................................... 22
    2.3.3 Source-Filter model............................................................................................................... 22
  2.4 PHONETICS................................................................................................................................ 23
  2.5 REPRESENTATION OF THE SPEECH SIGNAL.......................................................................... 25
    2.5.1 Fourier analysis..................................................................................................................... 25
      2.5.1.1 Discrete-time Fourier transform....................................................................................... 25
      2.5.1.2 Discrete Fourier transform............................................................................................. 25
    2.5.2 Cepstral analysis................................................................................................................... 26
  2.6 ANALOG TO DIGITAL CONVERSION OF SPEECH................................................................. 26
    2.6.1 Sampling............................................................................................................................... 27
    2.6.2 Quantization.......................................................................................................................... 27
      2.6.2.1 Number of channels......................................................................................................... 28
      2.6.2.2 Codec.............................................................................................................................. 28
      2.6.2.3 File formats.................................................................................................................... 28
  2.7 SPEECH RECOGNITION........................................................................................................... 28
    2.7.1 Isolated words....................................................................................................................... 28
    2.7.2 Connected words.................................................................................................................. 29
    2.7.3 Continuous speech................................................................................................................. 29
    2.7.4 Speaker recognition.............................................................................................................. 29
    2.7.5 General architecture of speech recognition system.............................................................. 29
    2.7.6 Challenges in speech recognition.......................................................................................... 31
    2.7.7 ASR softwares....................................................................................................................... 31
      2.7.8 Sphinx-4............................................................................................................................... 31
        2.7.8.1 Capabilities...................................................................................................................... 32
        2.7.8.2 Performance.................................................................................................................... 32
        2.7.8.3 Sphinx-4 architecture................................................................................................... 32
  2.8 THE TELEPHONE CHANNEL..................................................................................................... 33
  2.9 VoIP............................................................................................................................................. 33
  2.10 VoIP vs PSTN............................................................................................................................ 34
  2.11 WEB TELEPHONY................................................................................................................... 34
List of Figures

Figure 2-1: Waveform and spectrogram of the speech signal ‘Ericsson Research Luleå’.. 4
Figure 2-2: The human vocal organs, figure by Teachit [2]................................. 5
Figure 2-3: Source-filter model, figure by MIT OCW [4].................................... 6
Figure 2-4: Analog to digital conversion of Speech.............................................. 10
Figure 2-5: General architecture of speech recognition engine............................ 13
Figure 2-6: Sphinx-4 Decoder architecture......................................................... 16
Figure 3-1: Protocols used during Red5Phone call............................................. 22
Figure 3-2: Red5Phone system setup................................................................. 23
Figure 3-3: Original Red5Phone architecture..................................................... 24
Figure 3-4: Extended Red5Phone architecture with voicemail to text conversion interface................................................................. 26
Figure 3-5: Red5Phone with new message notification and the extended inbox with visual voicemail................................................................. 27
Figure 3-6: Voicemail widget, a web user is sending voicemail............................ 28
Figure 4-1: Performance of the configured ASR in test 1.................................... 31
Figure 4-2: Spectrograms of a speech signal sampled three different frequencies.... 33
Figure 4-3: Performance of the configured ASR in test 2.................................... 36
Figure 4-4: Results obtained from speech enhancement methods.......................... 39
## List of Tables

Table 2-1: Speech spectrum........................................................................................................ 4  
Table 2-2: Phonemes in American English.................................................................................. 8  
Table 2-3: Performance of Sphinx-4.......................................................................................... 15  
Table 3-1: Words pronunciation in CMUcmu 0.6 dict................................................................. 21  
Table 4-1: WER example........................................................................................................... 29  
Table 4-2: Speakers description involved in test 1................................................................. 30  
Table 4-3: Misrecognized words obtained during the recognition........................................ 32  
Table 4-4: List of use cases of Test 2......................................................................................... 35
List of Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASR</td>
<td>Automatic speech recognition systems</td>
</tr>
<tr>
<td>DFT</td>
<td>Discrete Fourier transform</td>
</tr>
<tr>
<td>DTFT</td>
<td>Discrete-time Fourier transform</td>
</tr>
<tr>
<td>HMM</td>
<td>Hidden markov model</td>
</tr>
<tr>
<td>IDFT</td>
<td>Inverse discrete Fourier transform</td>
</tr>
<tr>
<td>IPA</td>
<td>International phonetic alphabet</td>
</tr>
<tr>
<td>LTI</td>
<td>Linear time invariant</td>
</tr>
<tr>
<td>MERL</td>
<td>Mitsubishi Electric Research Labs</td>
</tr>
<tr>
<td>MFCC</td>
<td>Mel-frequency cepstral coefficients</td>
</tr>
<tr>
<td>MIT</td>
<td>Massachusetts Institute of Technology</td>
</tr>
<tr>
<td>NAT</td>
<td>Network address translation</td>
</tr>
<tr>
<td>PCM</td>
<td>Pulse code modulation</td>
</tr>
<tr>
<td>PSTN</td>
<td>Public switched telephone network</td>
</tr>
<tr>
<td>RTMP</td>
<td>Real-time messaging protocol</td>
</tr>
<tr>
<td>RTP</td>
<td>Real-time protocol</td>
</tr>
<tr>
<td>SIP</td>
<td>Session initiation protocol</td>
</tr>
<tr>
<td>TCP</td>
<td>Transmission control protocol</td>
</tr>
<tr>
<td>UCSC</td>
<td>University of California at Santa Cruz</td>
</tr>
<tr>
<td>UDP</td>
<td>User datagram protocol</td>
</tr>
<tr>
<td>VoIP</td>
<td>Voice over IP</td>
</tr>
<tr>
<td>WER</td>
<td>Word error rate</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Thesis objective

The objective of this thesis is to introduce a new service which is not available today in pc-based voice over IP (VoIP) services. The new service introduced is a web based inbox for voice mails being recorded in English. Received messages are transcribed automatically, converted to text and stored in an inbox. The speech recognition for this purpose has to be speaker-independent and the vocabulary must be quite big in order to facilitate continuous speech. It must also be prepared to handle distorted speech with background noise to some extent.

1.2 Problem Definition

Today all popular PC-based VoIP services are depending on a client software that must be installed explicitly by the user. The user must also register and log in before the user can be used which makes it difficult to integrate speech services into a browser environment. Moreover in most cases it is hectic and annoying for the user to go through the process of registration, downloading, and installation. Also there are situations when it is needed to see and read the voicemails instead of listening to them.

1.3 Hypothesis

Web based telephony can be helpful in coping with the current faced problems. The idea behind web based telephony is to enable telephony services inside a standard browser without any additional plug-ins. By combining web technology and traditional telephony services new exciting services can be created. Today web based telephony can be realized if the client is based on an existing and widely spread multimedia plug-in, such as Adobe Flash. Red5Phone is an open source Session initiation protocol (SIP) phone solution. It provides a client for the browser (Adobe Flash) and a SIP server application for Red5 (java). With Red5Phone and a Flash-enabled browser calls can be made or received with just a single click on the mouse button – no registration, no download. As said that new exciting services can be created in web based telephony. The new service could be a web based inbox to deal with the other problem.

1.4 Thesis Scope

The scope of this thesis work is to investigate, develop, implement and test a web based inbox for voice mails. The performance of speech transcription in practice is investigated taking into account different cases.
1.5 Thesis outline

Chapter 2 includes the relative theoretical background about speech and its basics. It also includes the basics of automatic speech recognition systems (ASR). Existing ASR softwares are also listed in this chapter. Theoretical study of SPHNX4 and Red5 phone are also incorporated in this chapter.

Chapter 3 explains the implementation methodology of the new service. It elaborates how Sphinx-4 was configured for the transcription of voicemails. Furthermore the architecture of the new introduced service is explained in detail.

Chapter 4 contains the performance evaluation and results. And in the end conclusion and future work are discussed in chapter 5.
CHAPTER 2  

Background

This chapter provides the reader a preliminary and essential background study of the terms, and methods needed in this thesis work. It is important to study this chapter in order to understand the overall thesis work up to its merits. It explains the speech basic and topics related to it. It also gives an overview of a speech recognition system. Speech recognition softwares that are in use are listed here. Moreover it also includes the study of Sphinx-4 and Red5Phone. Some light is thrown on VoIP and issues related to it.

2.1 The speech signal

Sound is produced when vibrations produced from an object move through a medium. There may be many different mediums responsible for transmitting sound waves, but the most familiar medium is air. In humans sound is taken as something opposite to silence. It is perceived as physiological sensation detected by human ears caused by multi fluctuations in the air. Human ears are sensitive enough to detect any kind of variation in air pressure within an ample range of frequencies. The speech signal is a complex, non-stationary, and non-periodic signal containing diverse instantaneous frequencies. Speech signal varies very slowly with time thus termed as quasi-stationary. Due to its quasi-stationary behavior the properties and statistics of speech signal are locally static over a short period of time, but exhibit difference from one local period to another [1]. Therefore the best way to study and characterize a speech signal is to spectrally analyze the speech signal samples taken at a very short interval of time. The speech signal can be best modeled mathematically by the following equation.

\[ s(t) = p(t) \otimes h(t) \]  \hspace{1cm} (2-1)

The above equation is among widely used modeling equations in speech related research work. It can also be termed as source filter model, which will be explained later in this chapter. \( s(t) \) in eq. (2-1) is the speech signal resulting from the convolution of speech signal \( p(t) \) with a filter \( h(t) \). An example waveform of the sentence “Ericsson Research Luleå” along with its spectrogram is given in the Figure 2-1.
Figure 2-1: Waveform and spectrogram of the speech signal ‘Ericsson Research Luleå’.

2.2 Sound spectrum

The speech spectrum is divided in three parts. The following Table 2-1 shows these spectrums along with their frequency ranges.

<table>
<thead>
<tr>
<th>Spectrum</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audible</td>
<td>20 Hz to 20,000 Hz</td>
</tr>
<tr>
<td>Infrasonic</td>
<td>&lt; 20 Hz</td>
</tr>
<tr>
<td>Ultrasonic</td>
<td>&gt; 20,000 Hz</td>
</tr>
</tbody>
</table>

Table 2-1: Speech spectrum

Audible frequency range is the range of frequencies which normal human ear can hear. Signals lying in the infrasonic range are rather felt than heard, whereas signals with frequencies greater than audible range are hardly heard by humans. They can cause serious damage to human auditory system.
2.3 Speech production

The information included in a spoken word (or speech to be more precise) is transmitted by the speech signal originated at the speaker. It is then perceived by a listener and the information is retrieved back from it. It is important to have a look on the process of speech production in humans. Having a clear understanding of the process it can be then modeled into filter-source model. Vocal organs are responsible for the production of speech in humans. These vocal organs are shown in the Figure 2-2.

Figure 2-2: The human vocal organs, figure by Teachit [2].

The production of speech starts as muscular movement (intentional) in lungs. Lungs and diaphragm are the main sources of energy needed to produce human sounds. Lungs store and process all the air needed for speech production. The spacing between the vocal folds is known is the glottis. The opening and closing of glottis puffs the air in an articulated way to produce different sounds. So during speech, the air produced at lungs thrusts its way aggressively through the windpipe. It then passes through the glottis towards vocal tract, pharynx and oral/nasal cavities. The vocal cords vibrate dynamically in many ways by the passage of air during the speech. The rate at which vocal cords vibrate is known as fundamental frequency. It is mainly determined by the mass and tension of vocal cords. After the sound is passed through the cavities, it is finally uttered by passive and active articulators.
2.3.1.1 Voiced Sounds

The sounds produced due to the vibration of vocal cords are called voiced sounds. The vibration rate (resonance frequencies) of vocal-tract is known as formant. Voiced sounds include all English vowels ([a], [e], [i], [o]) and [b], [d], [g], [v], [z].

2.3.1.2 Formant

Voiced sounds consist of a fundamental frequency (f0) and its harmonics produced by the vocal cords. The vocal tract modifies the quasi-periodic excitation signal produced at the fords causing formant band. In a spectrogram formants represent the energy concentration present in the phonemes of voiced sounds. Sounds can be much better identified by the formants and their alterations.

2.3.2 Unvoiced sounds

Sounds which are produced without involving the movement of vocal cords are known as unvoiced sounds. The articulatory organs in this case are used to create constrictions in the passage of air, and this constriction cause turbulence in the airflow resulting in unvoiced sounds. During the production of unvoiced sounds vocal cords remains relaxed. [p], [t], [k], [f], [s] and all other sounds falls in this category. It is important to know that unvoiced sounds usually possess broadband spectrum due to the absence of resonant peaks. In case of pure unvoiced sounds there is not any fundamental frequency in the excitation signal produced. Since there is not any fundamental frequency there will be no harmonics. The excitation produced in unvoiced sounds can be considered as white noise. Unvoiced sounds are more silent and less steady than the voiced ones.

2.3.3 Source-filter model

The traditional source-filter model consists of an excitation followed by a linear-varying filter. The excitation can be white Gaussian noise for unvoiced sounds or an impulse train for voiced sounds [3]. This model explains how the acoustic pulses produced at the glottis are shaped by the vocal-tract. Whereas glottis is the source and vocal-tract is the filter.
As seen from the above Figure 2-3 the excitation generator (source) gives input (Glottal airflow) to the filter. The train of glottal pulses enters the vocal tract also causing some radiations. Then the vocal tract along with the radiations acts as a linear time varying filter. And the frequency response of this filter when observed shows formants corresponding to the resonance frequencies. In this model the source and model are independent of each other. The filter transfer function (between glottis and the lips) is given as below

$$T_s(s) = \frac{P_m(s)}{P_h(s)}$$

where $P_h(s)$ in eq. (2-2) is the air turbulence resulting from the vocal-cords, and $P_m(s)$ is the net air pressure coming out of the lips.

### 2.4 Phonetics

For a proper pronunciation of words in any language a symbolic representation is needed. These representations are mostly language dependant. Phoneme is the smallest constructive unit of sound and every word is represented by a set of phonemes. Phonemes are nonfigurative entities and their pronunciation is subjective to many factors. Phones are described using two different alphabets i.e. the International Phonetic Alphabet (IPA) ARPAbet. In most cases like in American-English IPA is used. The following Table 2-2 lists phonemes in American English.

<table>
<thead>
<tr>
<th>IPA</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Λ</td>
<td>Luck</td>
</tr>
<tr>
<td>a:</td>
<td>Rather</td>
</tr>
<tr>
<td>æ</td>
<td>Hat</td>
</tr>
<tr>
<td>ø</td>
<td>Cinema</td>
</tr>
<tr>
<td>e</td>
<td>Bet</td>
</tr>
<tr>
<td>3: Turn</td>
<td>i Bit</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>i: Beat</td>
<td>ν Lock</td>
</tr>
<tr>
<td>ó Four</td>
<td>ó Would</td>
</tr>
<tr>
<td>u: Food</td>
<td>ai Eye</td>
</tr>
<tr>
<td>ao How</td>
<td></td>
</tr>
<tr>
<td>õ Rome</td>
<td>eœ There</td>
</tr>
<tr>
<td>ei Say</td>
<td>iœ Near</td>
</tr>
<tr>
<td>œi Coin</td>
<td>œœ Tourist</td>
</tr>
<tr>
<td>b Block</td>
<td>d Dare</td>
</tr>
<tr>
<td>f If</td>
<td>g Garden</td>
</tr>
<tr>
<td>h Hi</td>
<td>j Yes</td>
</tr>
<tr>
<td>k Cat</td>
<td>l Line</td>
</tr>
<tr>
<td>m Man</td>
<td>n Nose</td>
</tr>
<tr>
<td>ñ Ring</td>
<td>p Parcel</td>
</tr>
<tr>
<td>r Ramp</td>
<td>s Sample</td>
</tr>
<tr>
<td>ñ Shrimp</td>
<td>t Tortoise</td>
</tr>
<tr>
<td>ŋ Cheat</td>
<td>ò Think</td>
</tr>
<tr>
<td>δ Brother</td>
<td>v Five</td>
</tr>
<tr>
<td>w Window</td>
<td>z Zinger</td>
</tr>
<tr>
<td>ʒ Vision</td>
<td>dʒ Large</td>
</tr>
</tbody>
</table>

Table 2-2: Phonemes in American English.

It is important to have understating of phonemes because different spoken phonemes have different energy concentrations and they also lie in distinctive spectrums.
2.5 Representation of the speech signal

As described above speech signal is quasi-stationary signal whose properties vary with time. So it is a signal which is function of time. For proper analysis of speech signal and its properties it is validly represented in the frequency domain.

2.5.1 Fourier analysis

The essential idea of Fourier analysis is that every signal can be significantly represented as a series of sinusoidal waves. The time and frequency domains are alternative ways of representing signals. The Fourier transform relates the representation of a signal in these two domains. The speech signal though itself does not possess the properties of a perfect sine wave, but it can be represented as sum of smaller waves. With Fourier transform the original speech signal is converted into frequency domain and now it correspond to the frequencies were present in the original signal. The theme of representing a signal as combination of simpler functions is to better analyze the signal.

2.5.1.1 Discrete-time Fourier transform

Discrete-time Fourier transform (DTFT) is the class of Fourier transform in which Fourier transform of aperiodic, Discrete time signals is taken. With DTFT an aperiodic discrete signal can be represented in a continuous, periodic frequency spectrum. The DTFT is mathematically characterized by the following equation

\[ X(e^{jo}) = \sum_{n=-\infty}^{\infty} x[n]e^{-jwn} \]  

(2-3)

where

\[ x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{jo})e^{jwn} \, d\omega \]  

(2-4)

2.5.1.2 Discrete Fourier transform

In case of analog discrete function (signal) sampled at specific window and then extended periodically, the Discrete Fourier transform (DFT) is found. The DFT of the original signal can be related to the original signal by inverse DFT. DFT is very helpful in the spectral investigation of speech signals. The input of DFT is N number of samples of a time-domain sampled signal, and the output is frequency dependent signal coming out of amplitudes of the sines and cosines in the input signal. A signal of N sampled input points is transformed into sequence of M points by the DFT according to the following mathematical equation
\[ X[k] = \sum_{n=0}^{N-1} x[n] e^{-\frac{2\pi j k n}{M}} \] 

(2-5)

where

\[ x[n] = \frac{1}{M} \sum_{k=0}^{M-1} X[k] e^{\frac{2\pi j k n}{M}} \] 

(2-6)

Normally the number of input points N may not necessarily be equal to the output frequency points M.

### 2.5.2 Cepstral analysis

Cepstral analysis is the most significant among speech signal representations, because it is one of the most important components of front-end processing in ASR systems to obtain the acoustic features from a speech signal. To understand the concept of cepstral analysis it is necessary to look at the source-filter model of speech. Which term the speech signal as an output of a linear time invariant (LTI) system. The output is obtained when an input signal originating from sound source is convolved with an impulse response of a filter. Cepstral analysis is the process of de-convolution parameterizing the speech signal in terms of source-filter model. The speech signal according to source-filter model is represented in eq. (2-1). I.e. The excitation at glottis \( p(t) \) is convolved with filter \( h(t) \) to obtain speech signal \( s(t) \). Convolution in time domain is product in frequency domain. eq. (2-1) in frequency domain is given by

\[ S(f) = P(f)H(f) \] 

(2-7)

Now, the main concern of spectral analysis is to separate the spectral envelope and spectral details, and it is done by taking the log of eq.2-7

\[ \log(S(f)) = \log(P(f)) + \log(H(f)) \] 

(2-8)

Taking Inverse DFT of eq. (2-8) will separate the components of \( P(f) \) and \( H(f) \), and as a result of it cepstrum is obtained. The cepstrum obtained in eq. (2-8) are translated to a perceptual scale of pitches that is more realistic with human auditory systems i.e. Mel-scale using Mel-filter bank (filter weights). In state-of-the art ASR systems Mel-frequency cepstral coefficients (MFCC) are used for parametric representation of the speech signal.

### 2.6 Analog to digital conversion of speech

The analog speech signal should be digitized in order to be available for further processing in the computers. When a speaker speaks, the variations in air pressure vibrate the magnetic plates inside the microphone resulting in analog electric pulses.
These electric pulses are then digitized by the sound-cards in computers. The analog to digital conversion comprises of two main steps i.e. sampling and quantization, shown in Figure 2-4.

![Diagram of Analog to Digital Conversion](image)

**Figure 2-4: Analog to digital conversion of Speech**

### 2.6.1 Sampling

Speech signal is converted into a sequence of samples that are taken at a certain instant of time. Sampling of signal at a particular time means recording the amplitude of signal at that specific point of time. The number of samples taken per second defines the sampling rate which is measured in hertz. For an appropriate representation of a signal (and to avoid different errors) the sampling rate must follow Nyquist theorem. Mathematically

\[
F_s \geq 2F_{\text{max}}
\]

where

\[
F_s = \frac{1}{T_s}
\]

\(F_s\) is the sampling frequency, \(T_s\) is the sampling time, and \(F_{\text{max}}\) is the maximum frequency or bandwidth of the signal. In human speech the majority of the information lies within the range of 10 kHz, therefore to faithfully represent it 20 kHz sampling rate would be required. Microphone data is sampled at different sampling rates. Compact disks (cd’s) record the speech at a sampling rate of 44.1 kHz. The bandwidth in fixed telephone channels is limited to allow frequencies in the range of 4 kHz. Therefore 8 kHz (narrow-band) sampling rate is required in case of telephone speech.

### 2.6.2 Quantization

Every sample (measured amplitude) of the speech signal obtained during sampling must be stored as integers. The amplitudes are usually stored as 8 bit or 16 bit. For a telephone speech the samples are quantized to 16 bit PCM (pulse code modulation). The encoder described below (G.711) uses 14 bits from every sample and converts it to 8 bit log. The quantized data is then stored in various formats.
2.6.2.1 Number of channels

Channels in speech are trails for passing information in speech. It could be mono (single channel), stereo (2 channels), or channels greater than 2. To contain directional information in a speech two or more channels are required. Stereo speech signals require two channels and contain directional information.

2.6.2.2 Codec

Digitized speech is further compressed using different compression algorithms (codec). Speech codec are important in achieving low transmission rates in applications like mobile telephony and VoIP. The simplest among coding techniques is PCM. And the most important compression technique used in telephony is $\mu$-law and a-law (used by G.711). Other significant speech codecs are G722, GSM-EFR, AMR-NB, and AMR-WB.

2.6.2.3 File formats

There are number of standard file formats describing the speech, and used to store the speech file. These file formats contains headers. The famous file formats are wav (Microsoft), MP4, and OGG etc.

2.7 Speech recognition

The process of recognizing and understanding (in some cases) the speech is known as speech recognition. It is generally termed as ASR. It is a way of perception of the human speech by machines. In other words it is a mean of talking to the machines. The spoken speech signal is converted into a string of words using ASR. Speech recognition has broad applications encompassing different fields like health care, dictation, military purposes, telephone domain, command and control set-ups, and various embedded applications. There are various types of speech recognition depending upon its usage.

2.7.1 Isolated words

Isolated word recognition is based on the premise that the signal in a prescribed recording interval consists of an isolated word, preceded and followed by silence or other background noise [5]. In isolated words speech recognition system a single utterance is considered at a time. The recognizer expects to have silence before and after occurrence of every utterance. Its common use is single command or word response systems and is the simplest to implement. One of its applications is seen in voice-controlled mobile phones.

2.7.2 Connected words
In such systems the recognizer is capable of handling phrases spoken by a user. These phrases may contain more than a single word, but every word should be spoken clearly with short breaks between them. So that the words don’t run into one another.

2.7.3 Continuous speech

Continuous speech recognition systems are the hardest to implement because of a range of effects and acoustic conditions. Such recognizers are speaker-independent. In such systems the recognizer is made to handle continuous speech with words expected to run into one another. The short breaks are not compulsory for the user in such systems. And due to this the recognizer should be capable of using different methods to find the boundaries and end points. It is totally natural, where the users can speak as like as any other natural conversation. It is more like giving dictation to the computer. Majority of such systems are expected to handle large vocabulary (20,000~64,000) giving user a freedom to say anything like in natural speech.

2.7.4 Speaker recognition

It is the biometric identification of a speaker using speech. A speaker is recognized among the stored database of speakers based on the stochastic model of speakers created by the system using different features. Such systems are termed as speaker-dependant speech recognition systems.

2.7.5 General architecture of a speech recognition engine

The main components of a normal speech recognition system are given in the schematic shown in Figure 2-5. The input is a speech signal and the output is a string of words recognized by the system.

A typical ASR is composed of front-end and back-end. The front-end is responsible for taking the speech input and extracting features out of it. While the back-end contains the training data (language model, acoustic model, vocabulary) and performs the search for better match (recognition).

As seen from Figure 2-5 the speech signal is parametrically represented and features are extracted from it. This process of feature extraction involves many steps. Firstly the speech signal is pre-emphasized i.e. the energy content in higher frequencies is enhanced. As it is known that speech signal is quasi-stationary signal so the main area of interest is extracting the features of a smaller window of the speech signal. Rectangular and Hamming windows are used for windowing, but due to its better performance Hamming window is used. The windowed speech is then subjected to DFT resulting in information about energy content at different frequencies. The frequency bands are then mapped on mel scale and its log is taken.
The final stage of feature extraction is taking the inverse discrete-Fourier transform (IDFT) logged mel filter banks, resulting in MFCC. The spectral features are tended to be observations $Y$ and $W$ are the words in a word string. In the acoustic model probabilities of maximum likelihood of these observations are calculated. Probabilities are calculated for every single frame. So the acoustic model in overall is a probability vectors sequence. Due to the stochastic behavior of speech stochastic models are applied to find the best match in the available words. Mathematically it can be given as

Figure 2-5: General architecture of speech recognition engine.
\[ \hat{W} = \arg \max_{w \in \ell} P(W | Y) \]  

(2-9)

\( \ell \) is the space of all possible words string available. And using Bayes’ rule eq. (2-9) becomes

\[ \hat{W} = \arg \max_{w \in \ell} \frac{p(Y | W) p(W)}{p(Y)} \]  

(2-10)

or

\[ \hat{W} = \arg \max_{w \in \ell} p(Y | W) p(W) \]  

(2-11)

The search phase makes use of the acoustic model, lexical model based on HMM, and language model resulting in the best match of words.

2.7.6 Challenges in speech recognition

Speech recognition is a challenging task. There are many challenges which should be addressed. Among such concerns affecting the performance of ASR are noisy background, complexity of spoken language, continuous speech with large vocabulary, channel inconsistency, speaker variability, accent of the speaker (and dialects), realization of words, speed of the speech etc.

2.7.7 ASR softwares

The ASR software which are available for free include XVoice, CVoice, open mind speech, GVoice, ISIP, CMU Sphinx, Ears, NICO etc. Commercial ASR software includes IBM Viavoice, Vocalis speechware, Babel Technologies, Speechworks, Nuance, Abbot, Entropic, and etc. Nuance Dragon Naturally speaking 10 is supposed to be one of the best commercial ASR software.

2.7.8 Sphinx-4

Sphinx-4 is an open-source java based state-of-the-art ASR system. It was developed by Sphinx group which was a joint venture of Carnegie Mellon University, Sun Microsystems Labs, Mitsubishi Electric Research Labs (MERL), Hewlett Packard, with contributions from University of California at Santa Cruz (UCSC) and Massachusetts Institute of Technology (MIT) [6].

2.7.8.1 Capabilities
Sphinx-4 has implemented standard speech recognition techniques along with improved methods and has a variety of capabilities. It can recognize discrete and continuous speech both in live and batch modes. It has a pluggable framework and can be configured for applications with different needs. It has pluggable front end, language, and acoustic models. The search management is also generalized and pluggable. It also has some standalone tools.

2.7.8.2 Performance

The performance of Sphinx-4 as given by the Sphinx-4 team is given in the following Table 2-3 [6].

<table>
<thead>
<tr>
<th>Test</th>
<th>S4 WER</th>
<th>Vocabulary Size</th>
<th>Language Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>TI46</td>
<td>0.168</td>
<td>11</td>
<td>isolated digits recognition</td>
</tr>
<tr>
<td>TIDIGITS</td>
<td>0.549</td>
<td>11</td>
<td>continuous digits</td>
</tr>
<tr>
<td>AN4</td>
<td>1.192</td>
<td>79</td>
<td>trigram</td>
</tr>
<tr>
<td>RM1</td>
<td>2.88</td>
<td>1,000</td>
<td>trigram</td>
</tr>
<tr>
<td>WSJ5K</td>
<td>6.97</td>
<td>5,000</td>
<td>trigram</td>
</tr>
<tr>
<td>HUB4</td>
<td>18.756</td>
<td>60,000</td>
<td>trigram</td>
</tr>
</tbody>
</table>

Table 2-3: Performance of Sphinx-4

- WER - Word error rate (%) (lower is better)

Sphinx-4 will be configured for continuous speech with maximum vocabulary size (60,000). The WER for HUB4 and vocabulary size of 60,000 is 18.756 which is quite reasonable.

2.7.8.3 Sphinx-4 Architecture

The Sphinx-4 framework has been designed with a high degree of flexibility and modularity. Figure 2-6 shows the overall architecture of the system. Each labeled element in Figure 2-6 represents a module that can easily be replaced, allowing researchers to experiment with different module implementations without needing to modify other portions of the system [7].
As seen from Figure 2-6 Sphinx-4 has three main parts i.e. the front-end, the decoder, and the linguist. The front-end is responsible for taking the input speech and converting it into parametric representation (features extraction) which will further be processed. The linguist creates search graphs with the help of language model, dictionary, and acoustic model. These graphs along with the features extracted are used by the search manager in the decoder to give best matching results.

2.8 The telephone channel

Channel in communication system refers to the mean (medium) by which a sender sends information to the receiver. Telephone channel is a fully-duplex channel i.e. communication channels which allow transmission in both directions at the same time. The telephone channel is band-limited allowing frequencies upto 4 kHz (sampling rate of 8 kHz). 256 quantization levels are used for which 8 bits per level are consumed. So with a bit rate of 8 bits and sampling rate of 8 kHz the capacity of telephone channel becomes 64 kilo bits per second (kbps).

2.9 VoIP

VoIP stands for voice over internet protocol. VoIP basically involves the process of transmitting the digitized speech using packets over the internet. A broadband internet connection is required for VoIP. It is a packet-switched service. The packet transmission in VoIP involves certain protocols and compression techniques. The important components of VoIP are transmission (of the data) and signaling (for sessions) mechanisms. The analog-to-digital conversion involved in VoIP has been explained before in this chapter. As a part of audio coding codec are used to achieve the compression. Different codec encode at a different data rate range. For example, AMR-NB (4.75 – 12.2 kbps), and G.711 (64 kbps). Wide-band (50 Hz~7 kHz) codecs like AMR-WB (6.6 – 23.85 kbps) or G.719 are of great interest in VoIP. RTP/UDP/IP
protocols are deployed by VoIP for media packet transmission. Signaling protocols enlist H.323 (superseded by MGCP), H248 (Megaco), and SIP. VoIP is an outweighed technology with a mounting growth rate.

2.10 VoIP vs PSTN

VoIP has been welcomed by individuals and enterprises. It is rising as a strong competitor to Public Switched Telephone Network (PSTN). The most important edge that VoIP has over PSTN is the low costs since one shared network for both voice and data can be used. VoIP is simpler and doesn’t need much or complex hardware. With a DSL internet connection free pc to pc calls could be made, and the charges to call to a phone are very minimal as compared to calling from a PSTN. Another advantage of VoIP is portability. It is just like other internet service. And this portability makes it capable of offering new features as the technology keeps evolving. Along with advantages it has some disadvantages. The voice quality may degrade for certain reasons (related to packet network transport) like delay, jitter (variations in the delay of packets), and packet loss (packet totally dropped by the network). Accessibility and reliability can be addressable priorities for VoIP traffic. Whereas Accessibility is: can a user always connect whenever needed? I.e. internet is accessible and SIP service is available. And reliability is: does the internet connection disconnect, provide enough bandwidth available and is the delay low enough? These issues for instance depend upon the service the ISP provides. Emergency calls are also an issue in case a power-outage is faced. Another problem is portability: the caller ID (phone number) can not be mapped to a specific geographical location.

2.11 Web telephony

Web telephony is activating and making Calls from a web environment. It is not possible to activate and make web calls with current web technologies; instead it relies on a plug-in like Adobe Flash. There is no need to download or install certain software (like many VoIP softwares). Everything is done within the browser. The calls could be either made to other web-based clients or a telephone numbers via gateways.

2.12 Red5Phone

Red5 [8] is a free and open source flash media server implemented in java. It is capable of audio/video streaming/recording, live stream publishing, and remoting. Red5Phone [9] is an open source SIP application for the Red5 server. It has two parts. A flash based soft phone and a java based server application. The server part of Red5Phone consists of a Red5 application called SIP which uses MjSIP [10] as the SIP user agent.
2.13 Speech recognition for voicemail

Voicemail can be termed as similar to email but here voice messages are conveyed. The messages are stored on a centralized system that can be accessed by the user through the operator. In the case of speech recognition for voicemails the stored voicemails are recognized and converted into texts. It could be termed as visual voicemail where the massages are readable. Speech recognition for voicemails is a tricky job for many reasons such as the limited bandwidth of telephone speech, distorted speech, background noise, and accents.

2.14 State-of-the art in speech recognition for voicemail

There are solutions available to transcribe voicemails. And several telecommunication companies have integrated their voicemail systems with transcription systems, like T-mobiles in USA. The market leading in speech transcription Nuance has launched its voicemail to text service that promises a high accuracy (99.9%). The reason for such high accuracy along with the improved ASR technology is the team of 3000 in-house transcriptionists. So, it makes Nuance’s voicemail to text as a semi-automatic system. The transcriptionists are involved in fixing the transcription errors and forwarding the correct transcription. Nuance’s voicemail to text service is based on famously Dragon Naturally speaking and can be easily integrated with the carrier’s system. Google voice is expected to be launched soon which will also offer transcription of voicemails. Another famous Spinvox (now a subsidiary of Nuance) provides its services to carriers (wireless, fixed and VoIP), unified communications, and web 2.0 environments. nFinity, ipadio, and Skype uses Spinvox services. GotVoice is another service provider in the market which offers voicemail transcription for individuals, business users, and partners. VoiceCloud claims to use a patent-pending technology they use for transcription of voicemails. Callwave offers widgets for the transcription.
This chapter explains the implementation methodology of the new service. The first section elaborates the configuration of Sphinx-4, whereas the later part covers the architectural details of the new service.

3.1 Configuration of Sphinx-4 for the system

The main goal was to have a recognition system that will be suitable for transcription of speaker independent, continuous, and a large vocabulary speech. Sphinx-4 has three main blocks in its framework i.e. the front end, the decoder, and the linguist. The properties of every block were tuned at different values. These values were chosen among the range of values that were successfully tested by the Sphinx-4 team. The values which gave best performance and transcription results were used in its final configuration. The configuration of Sphinx-4 is step-wise explained below.

3.1.1 Frequently used properties

These properties of the configuration file are those which are frequently used

- Absolute Beam Width

  It limits the recognizer to recognize maximum possible suitable utterances at word level. Absolute beam width guides one of the active list factories of Sphinx-4. It helps decoder discard the least matching hypothesis from the list available for each frame. Its value was set to 500 to meet the maximum utterances among a conversation.

- Relative Beam Width

  Relative beam width is a threshold to be used at the end of decoding of every frame. It relates the best match with every possible match of utterances. Its value was set to $10^{-120}$.

- Word Insertion Probability

  During the recognition there are lattice paths available for words, silence etc. word insertion probability is used to set the arc at the beginning of a new word at a lattice. Its value was set to 0.7.

- Silence Insertion Probability

  Like word insertion probability this tunable property will set the arc of a lattice at the beginning of ever silence phoneme. Its value was set to 0.1.

- Language Weight
This value decides the contribution of Language Model towards the overall score. Its value was set to 10.5.

3.1.2 The search manager

The search manager is a component of the Decoder block of Sphinx-4. It performs the search for best results using features from front end and scores from the linguist. Word Pruning Breadth First Search Manager was used.

- Log Math

This property of the search manager sets all the probabilities and scores in log Math log domain while the best score is find by the search manager.

- Pruner

The pruner is used at every frame level to prune the less probable lattice, both for absolute beam width and active beam width. Its value was set to trivial Pruner.

- Scorer

The scores of search in the decoder are managed by scorer. Its value was set to threaded Scorer which uses a single token thread at a time.

- Active List Manager

It contains active list factories used by the search manager during the pruning of every search state type.

3.1.3 The linguist

The linguist is one of the three main blocks of Sphinx-4 framework. It contains Language model, Acoustic model, and dictionary and uses them to create search graphs. Since the system was needed to be configured for larger vocabulary, Lex tree linguist was used. That is the one available for larger vocabulary use. The language model, acoustic model, and dictionary are main properties of the Linguist block.

- Acoustic model

Acoustic model is one of the important components of the Linguist block. It must be available to the decoder in order to perform recognition. Acoustic model contains statistical representations of phonemes according to their corresponding pronunciations. Acoustic model must be trained on a real speech data along with the real correct transcription of that speech data. Sphinx-4 is shipped along with Hidden markov model (HMM) based acoustic model that is trained on a larger speech data
and covers a vocabulary of 64000 words. This acoustic model was used. HUB4 acoustic model was trained on 140 hours of broadcast news speech data that was sampled at 16 kHz.

- **Language model**

A statistical language model assigns probabilities to a set of words. It captures the language properties and on the basis of that it predicts the next word in the sequence. HUB4 Language model was used which is shipped along with Sphinx-4 and it meets the requirement of a larger vocabulary. It is a trigram language model. Trigram language model predicts a word on the basis of conditional probability i.e.

\[ p(w_3|w_1,w_2) \]

This means that in a sequence of words in a certain language the word \( w_3 \) will immediately follow sequence of words \( w_1,w_2 \).

- **Dictionary**

The dictionary contains the representation of words according to their phonemes. Sphinx-4 has its own dictionary by the name ‘CMUdict’. The version of this dictionary used was 0.6. This dictionary is based on 39 phonemes. And example from the dictionary is given below [11] in the Table 3-1.

<table>
<thead>
<tr>
<th>Phoneme</th>
<th>Example</th>
<th>Translation</th>
</tr>
</thead>
<tbody>
<tr>
<td>AA</td>
<td>odd</td>
<td>AA D</td>
</tr>
<tr>
<td>AE</td>
<td>At</td>
<td>AE T</td>
</tr>
<tr>
<td>AH</td>
<td>Hut</td>
<td>HH AH T</td>
</tr>
<tr>
<td>AO</td>
<td>Ought</td>
<td>AO T</td>
</tr>
<tr>
<td>AW</td>
<td>Cow</td>
<td>K AW</td>
</tr>
<tr>
<td>AY</td>
<td>Hide</td>
<td>HH AY D</td>
</tr>
<tr>
<td>B</td>
<td>Be</td>
<td>B IY</td>
</tr>
<tr>
<td>Ch</td>
<td>Cheese</td>
<td>CH IY Z</td>
</tr>
</tbody>
</table>

Table 3-1: Words pronunciation in CMUcmu 0.6 dict.

**3.1.4 The front end**

The front end is an important module of Sphinx-4 and it extracts the features out of a speech input. The front end can be configured to extract different features. The module consists of several communicating blocks, each with an input and an output. Each block has its input linked to the output of its predecessor [12]. The front end can be configured for both live speech recognition and batch mode speech recognition. In this case it was configured to be used in batch mode, and to extract MFCC from the speech input.
After the configuration of transcription system it was integrated with the voicemail architecture of Red5Phone. The detailed description is given in the coming sections.

### 3.2 Red5Phone

As telephony service was to be implemented into a web environment, it was needed to use a client-server web based system. So it required a flash based client and a server. Red5Phone provides a Flash based client and a server with SIP functionality, which could be extended for the transcription of voicemail.

### 3.3 Protocols

The protocols involved in the application are given in the following Figure 3-1.

![Figure 3-1: Protocols used during Red5Phone call.](image)
It is seen from the Figure 3-1 that the communication between Red5 server and a red5Phone client is using Adobe’s proprietary Flash protocol Real-Time Messaging Protocol (RTMP). This protocol is used for both the control of sessions and for the transport of media. The Red5 server always transcode and packetize the media sent by Red5Phone client into real-time protocol (RTP) packets. These packets are then transported to the client that can be a Red5Phone client or a SIP client. In case of Red5Phone client the packets must be depacketized and transcoded before it can be delivered to the client. In the case of a SIP client, it connects to the Red5 server via a SIP server. The sessions are then controlled using SIP protocol.

### 3.4 Setup

The recommended set up of Red5Phone system is given below in Figure 3-2, and this was the set up used.

![Diagram of Red5Phone system setup](image)

Figure 3-2: Red5Phone system setup.
The above setup is to run SIP server on one computer and Red5Phone server on another computer. This way the SIP traffic between the two servers can be caught and can be easily troubleshooting. It is possible to run both servers on a same computer but that was not tested. A flash based Red5Phone client or a standard SIP client such as X-LITE [13] can be used with the service.

3.5 Architecture

3.5.1 The original Red5Phone architecture

The original Red5Phone architecture is given in the following Figure 3-3 [9].

Figure 3-3: Original Red5Phone architecture.
As seen from Figure 3-3, when the Red5Phone connects to the Red5 Application it invokes the ‘open’ method for the connected user. The Red5 application uses a SIPManager to create a SIPUser object and an RTMPUser object for that user. The SIPUser object then contacts the SIP server and the user is registered at the server. In case of Red5Phone client making a call, it connects to the Red5 application and invokes a ‘call’ method. The SIPUser object created for the user will then contact the SIP server to start an outgoing call through SIP. After the call is started the Red5 application will then inform the client of what audio stream names it should use to publish audio from the microphone and to send to the speaker.

Red5Phone only supports 8000 Hz G.711 μ-law codec. The Red5 Application decoded the incoming RTP packets and changes the sample rate from 8000 Hz to 11025 Hz. The processed audio at the new sampling rate is then published by the RTMPUser object using the same stream name as the Red5phone client is playing.

The audio sent by the microphone must be transcoded before it can be delivered to the SIPUser object, because Adobe Flash Player 9 does not support any other codec than NellyMoser for recording. The transcoding from NellyMoser to G.711 μ-law takes place between the RTMPUser and the SIPUser.

An incoming call follows the same pattern where the RTMPUser object notifies the Red5Phone client. If the call is accepted by the user, the audio streams are set up in the same way as they were for an outgoing call. The original architecture is designed in such away that it can handle multiple calls and each telephone call requires two user RTMP connections and four audio streams.

### 3.5.2 Extended Red5Phone Architecture

The Red5Phone original architecture was previously changed to include voicemail functionary. Moreover the sampling rate was changed from 8000 Hz to 11025 Hz. Now the architecture was further extended with the new voicemail transcription functionality. The new RTMPUser class has a connection to the database where SIP credentials and SIP-server addresses are stored. For every new message a new post is created in the voicemail database with a link to the recorded speech file. The voicemail transcription functionality has been added to VMUser class. VMUser class works in the same way as SIPUser class. The extended VMManager class is responsible for creating and destroying the VMUser objects for connected users.

When a new voicemail is recorded in the RTMPUser class and written to file and saved on the desk, the transcription system is invoked within the extended RTMPUser class. The speech file is read from the file within RTMPUser class, and transcribed. The transcription is written to a text file and saved to the same disk location.

The extended architecture is given in the Figure 3-4.
3.6 **Ports and tunneling**

Web based telephony service must be able to establish and maintain network connections that can transverse network address translation (NAT) gateways and firewalls. 1925 is the default port for RTMP protocol. The Red5 server is also supporting the RTMP protocol which uses port 80 as a tunnel for RTMP messages, if port 1925 is closed.

The embedded web server (Jetty) in Red5Phone server is reachable at port 5080. The recommended browser to connect to this port is Mozilla Firefox.
3.7 Red5Phone clients

3.7.1 Red5Phone client

The Red5Phone client is a Flash based soft phone that connects to the server using a proprietary session control protocol. The client application is written in ActionScript 3 and the user interface is defined in MXML. Previously the client has been modified to include voicemail inbox. The extended inbox has now the option to display the transcription of the voicemail audio messages. So, now in the inbox choices of play, read, and delete the messages are available. A notification icon is shown for every received message.

![Figure 3-5: Red5Phone with new message notification and the extended inbox with visual voicemail.](image)

The above Figure 3-5 shows the client with notification of a new arrived message. The extended voicemail inbox can also be seen. A transcription of a voicemail message ‘I was trying to help you with the problems you had last year’ is also displayed.

3.7.2 Voice mail ‘widget’

The voicemail widget implemented in flash before is shown in the Figure 3-6. It can be used to record a voicemail with just a single click – no registration, no download. This widget uses the Red5Phone server interface to realize the voicemail service.
3.8 Transcription delay

The time taken to transcribe the arrived voicemail audio message by the system was noticed. It varied with the length of the voicemail message. But the normal average time calculated was greater than 5 times the length of the speech file. For example, for a voicemail message of 6 seconds the transcription system needed 35 seconds to transcribe the message and make the text available to the client.
CHAPTER 4 Performance evaluations

4.1 Performance Evaluation of the transcription system

When the transcription system (Sphinx-4) was configured with an extended vocabulary, language (HUB4) and acoustic (HUB4) models, multiple tests were performed to evaluate its performance. Test1 was performed with speech files (attained) from an online open source Voxforge speech corpus [14]. While test2 was performed using speech files from an Ericsson internal database. Total numbers of speech files used were 9 (3 speakers, 3 speech files each). WER is used as a standard to evaluate the performance. The correct transcription of wave files is called Reference transcription whereas the sequence recognized by the transcription system is called as hypothesized word sequence. WER is based on the difference between these two transcriptions. More precisely WER is the number of word substitutions (S), word deletions (D), and word insertions (I) needed in the Reference transcription to make it look-alike the Hypothesized transcription. WER is calculated as follows:

\[
WER = 100 \times \frac{S + D + I}{N}
\]

(4-1)

It must be noted that WER can exceed 100% in some cases because equation (1) also has insertions, so the total number of words in Hypothesized word sequence might be greater than the number of words \( N \) in the correct transcription. To understand WER better the following example in Table 4-1 can be referred.

<table>
<thead>
<tr>
<th>Hypothesized</th>
<th>He</th>
<th>added weight</th>
<th>had</th>
<th>a</th>
<th>Velocity of the</th>
<th>fifteen</th>
<th>mph</th>
<th>*</th>
<th>*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td>The</td>
<td>added weight</td>
<td>had</td>
<td>a</td>
<td>Velocity of</td>
<td>*</td>
<td>fifteen</td>
<td>miles</td>
<td>per</td>
</tr>
<tr>
<td>Operation</td>
<td>S</td>
<td></td>
<td></td>
<td></td>
<td>I</td>
<td></td>
<td>S</td>
<td>D</td>
<td>D</td>
</tr>
</tbody>
</table>

Table 4-1: WER example.

It is obvious from the example that there are 2 substitutions, 2 deletions, and one insertion needed in the Reference transcription to make it look-alike the hypothesized transcription. And WER in this example as calculated from (1) is 45.45%.

The test result’s tables also contain misrecognized words along with its correct replaceable words. Misrecognized words are supposed to be the words returned by the transcription system, which are more close to the correct words. And are more likely pronounced in a similar way as the correct words are pronounced. It will be discussed more in the test results.
4.2 Test1

In test1 the speech files were taken from Voxforge speech corpus. There were three speakers listed here as A, B, and C in the test results tables. Three speech files of each user were subjected to the test. Each speech file had a different spoken sentence. The sentences were of varying complexity. There were three use cases in this test. In case 1 speech samples sampled at 16 kHz were used. In case 2 the sampling rate of speech files was 8 kHz. While in case3 the sampling rate was again 16 kHz. Results from each user case are discussed separately.

4.2.1 Preparation of Data

Speech data was prepared for each use case. As, speech recognition is preceded by the process of data preparation [15]. The description of speech files downloaded from Voxforge speech corpus and the speakers A, B, C is given in the following Table 4-2.

<table>
<thead>
<tr>
<th>speaker</th>
<th>Username</th>
<th>gender</th>
<th>Age range</th>
<th>Language</th>
<th>Dialect</th>
<th>File type</th>
<th>Sample rate format</th>
<th>Bit rate</th>
<th>No of channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Unbridledrage</td>
<td>Male</td>
<td>Adult</td>
<td>English</td>
<td>American</td>
<td>wav</td>
<td>16000 HZ</td>
<td>16 bits</td>
<td>1</td>
</tr>
<tr>
<td>B</td>
<td>Aaron</td>
<td>Male</td>
<td>Adult</td>
<td>English</td>
<td>New Zealand</td>
<td>wav</td>
<td>16000 HZ</td>
<td>16 bits</td>
<td>1</td>
</tr>
<tr>
<td>C</td>
<td>Krellis</td>
<td>Male</td>
<td>Adult</td>
<td>English</td>
<td>American</td>
<td>wav</td>
<td>16000 HZ</td>
<td>16 bits</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 4-2: Speakers description involved in test 1.

**Case 1:** In this case the sampling frequency of the speech files was 16 kHz and the bits per sample were 16 bit.

**Case 2:** In this case 16 kHz 8 bit samples were used. Adobe Audition was used to down sample 16 KHZ files.

**Case 3:** The 8 kHz 16 bit speech files used in Case 1 were up sampled using Adobe Audition and then used in this case.

4.2.2 Results

The results of each use cases are given in a separate test results table. Each table contains reference transcription, hypothesized word sequence, substitutions, insertions, word error rate, and misrecognized words. The average WER calculated from the test result tables is shown in the Figure 4-1. And the performance of the
transcription system can be evaluated accordingly in the three cases. The lesser the WER, the better the performance.

![Average WER %](image)

Figure 4-1: Performance of the configured ASR in test 1.

### 4.2.2.1 Case 1

As seen from the chart the transcription system performs it’s best for the input speech files sampled at 16 kHz 16 bit. The WER of 16.63% in this case is lower than the WER of 18.75% [6] which was obtained by the Sphinx-4 team when they ran the regression tests on Sphinx-4 configured with HUB4 model. It is because there were fewer samples used for the test in this case. But if the test is performed with a greater number of samples, then apparently the WER will occur in the range of 18.75%. the best transcription result as seen from the chart are obtained in this case, because the transcription system was configured with HUB4 Acoustic model. And HUB4 acoustic model was trained using speech data sampled at 16 kHz.

It is noticed from the test results tables that there are misrecognized words in most of the transcriptions. Misrecognized words can be taken as those which do not have a major contribution in changing the perception of a whole sentence. And mostly these are the homophones. Homophones are the words which are spelled differently but pronounced almost similarly. For example some of the misrecognized words from the table along with their corresponding phonemes are given below in Table 4-3.
Table 4-3: Misrecognized words obtained during the recognition.

<table>
<thead>
<tr>
<th>Word</th>
<th>Sun</th>
<th>Saud</th>
<th>Men</th>
<th>Mean</th>
<th>Bored</th>
<th>Border</th>
<th>Spink</th>
<th>Stink</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phoneme</td>
<td>S ah n</td>
<td>s a od</td>
<td>M eh n</td>
<td>M iy n</td>
<td>B ao r d</td>
<td>B ao r d er</td>
<td>S p ih ng k</td>
<td>S t ih ng k</td>
</tr>
</tbody>
</table>

Now as it is seen that these words have matching sets of phonemes, and it is very much possible that a feature number is allowed to be used by more than just one phoneme. So during the recognition when the features are matched with the paths in the search graph, the path with highest score will be selected by search manager. And it could be ambiguous due to the statistical data of occurrence of a certain word in the language model. And as the graph also represent sequence of all possible phonemes, so the features are matched with the graph containing misrecognized words. Also the reason for this is Word boundary ambiguity, which occurs when the phones are combined to form words in more than one unique way. So with different grouping of phones a certain ambiguity in the system appears which ultimately decreases the recognition rate.

4.2.2.2 Case 2

From the chart it is witnessed that the performance degrades to a reasonable extent (40.38%) when 16 kHz down-sampled 8 kHz speech files were used. The main reason is that the acoustic model data was 16 kHz whereas the test speech files were 8 kHz. If the system is configured with an acoustic model trained with 8 kHz then the performance will for sure increase. Till now the Sphinx-4 team has been able to train one such acoustic model on 8 kHz data, but it has a very limited vocabulary of five thousand max.

Now in Figure 4-2 there are spectrograms of three speech waveforms i.e. 16 kHz, 8 kHz (16 kHz down-sampled), and 16 kHz (8 kHz up-sampled). It is obvious from the spectrogram of 16 kHz signal that its major spectral content lies in the frequency range of 0-8 kHz. So during recognition the Acoustic model will look for matching the patterns in this range. And when this signal of 16 kHz is down-sampled to 8 kHz the spectral content between 4-8 kHz is lost, as now in the spectrum of 8 kHz signal the major spectral content lies in the range of 0-4 kHz. There is still a fair recognition rate because if the spectrum of 16 kHz signal is analyzed, it could be sub divided into two frequency bands i.e. 0-4 kHz and 4-8 kHz, where the major spectral content lies in 4-8 kHz band. So a fair amount of patterns are recognized are matched within this spectral range.

4.2.2.3 Case 3

When the 8 kHz samples were up-sampled to 16 kHz, the performance increased to a slight extent and WER is reduced to 30.53 %. This apparently is because now the input speech signals tend to be of 16 kHz matching the training data of the acoustic
model. In practical up-sampling 8 kHz sample to 16 kHz does not change the major spectral content of the signal at all. As it is seen from the spectrogram of 16 kHz (8 kHz up-sampled) speech signal in Figure 4-2. The spectrogram now shows that the major patterns will be matched with the patterns of those in the band of 0-4 kHz. It is noticed from the spectrum that there is also a flat spectrum (almost equal content for the whole frequency spread) of the band 4-8 kHz. It is important to know that up-sampling would not create or rebuild in any way the missing patterns of 4-8 kHz band.

If the acoustic model is trained with real 8 kHz speech data, and then use the test data sampled at 8 kHz, the performance will greatly increased. And the WER will be far less than that in this case now, but it will still be greater than that obtained in case 1.

Figure 4-2: Spectrograms of a speech signal sampled three different frequencies.
4.3 Test 2

The speech files used in test2 were the one obtained from the Ericsson speech database. Three speech files of three different speakers were used in each case. The speakers in the transcription results tables for this test are listed as 1, 2, and 3. The description of speakers is given below.

**Speaker 1 (f2):** Denise from Vancouver, Canada: Light voice, well articulated, natural intonation.

**Speaker 2 (m1):** Ed from Dublin, Ireland: Medium dark voice, well articulated, slight Irish accent, natural intonation.

**Speaker 3 (m3):** Colm from Dublin, Ireland: Rather dark voice, well articulated, frequently vocal fry at sentence endings.

4.3.1 Preparation of test data

The speech data used for the test was prepared first. There were 22 rigorous use cases. ITU-T STL2000 tools, AMR tools, Sox tools, Speex tools were used in the process. There were use cases having clean speech files, speech files with different sampling rates, speech files corrupted with different background noises at different levels, speech files encoded with different encoders, and speech files having channel errors. The background noises were of two types. One was highway noise and other was food court noise. The speech files were corrupted with noise at two different levels i.e. -9 db and -18 db. The use cases along with their details are given in the following Table 4-4.

<table>
<thead>
<tr>
<th>Condition</th>
<th>BGN level/type</th>
<th>Codec configuration</th>
<th>Channel Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>Clean</td>
<td>Direct 16 kHz (8 kHz up-sampled)</td>
<td></td>
</tr>
<tr>
<td>C2</td>
<td>Clean</td>
<td>Direct 16 kHz (48 kHz down-sampled)</td>
<td></td>
</tr>
<tr>
<td>C3</td>
<td>BGN Highway - 9db</td>
<td>Direct 16 kHz (8 kHz up-sampled)</td>
<td></td>
</tr>
<tr>
<td>C4</td>
<td>BGN Highway - 18db</td>
<td>Direct 16 kHz (8 kHz up-sampled)</td>
<td></td>
</tr>
<tr>
<td>C5</td>
<td>BGN Food court - 9db</td>
<td>Direct 16 kHz (8 kHz up-sampled)</td>
<td></td>
</tr>
<tr>
<td>C6</td>
<td>BGN Food court - 18db</td>
<td>Direct 16 kHz (8 kHz up-sampled)</td>
<td></td>
</tr>
<tr>
<td>C7</td>
<td>BGN Highway - 9db</td>
<td>Direct 16 kHz (48 kHz down-sampled)</td>
<td></td>
</tr>
<tr>
<td>C8</td>
<td>BGN Highway - 18db</td>
<td>Direct 16 kHz (48 kHz down-sampled)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Use Case</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>C9</td>
<td>BGN Food court - 9db</td>
<td>Direct 16 kHz (48 kHz down-sampled)</td>
<td></td>
</tr>
<tr>
<td>C10</td>
<td>BGN Food court - 18db</td>
<td>Direct 16 kHz (48 kHz down-sampled)</td>
<td></td>
</tr>
<tr>
<td>C11</td>
<td>BGN Highway - 9db</td>
<td>16 kHz (8 kHz up-sampled), Speex encoded without noise suppression</td>
<td></td>
</tr>
<tr>
<td>C12</td>
<td>BGN Food court - 18db</td>
<td>16 kHz (8 kHz up-sampled), Speex encoded without noise suppression</td>
<td></td>
</tr>
<tr>
<td>C13</td>
<td>BGN Highway - 18db</td>
<td>16 kHz (48 kHz down-sampled), Speex encoded without noise suppression</td>
<td></td>
</tr>
<tr>
<td>C14</td>
<td>BGN Food court - 9db</td>
<td>16 kHz (48 kHz down-sampled), Speex encoded without noise suppression</td>
<td></td>
</tr>
<tr>
<td>C15</td>
<td>BGN Highway - 9db</td>
<td>16 kHz (8 kHz up-sampled), Noise suppression (Speex)</td>
<td></td>
</tr>
<tr>
<td>C16</td>
<td>BGN Food court - 18db</td>
<td>16 kHz (8 kHz up-sampled), Noise suppression (Speex)</td>
<td></td>
</tr>
<tr>
<td>C17</td>
<td>BGN Highway - 18db</td>
<td>16 kHz (48 kHz down-sampled), Noise suppression (Speex)</td>
<td></td>
</tr>
<tr>
<td>C18</td>
<td>BGN Food court - 9db</td>
<td>16 kHz (48 kHz down-sampled), Noise suppression (Speex)</td>
<td></td>
</tr>
<tr>
<td>C19</td>
<td>Clean</td>
<td>AMR Wideband Loss Free</td>
<td></td>
</tr>
<tr>
<td>C20</td>
<td>Clean</td>
<td>AMR Wideband 1% Loss</td>
<td></td>
</tr>
<tr>
<td>C21</td>
<td>Clean</td>
<td>AMR Wideband 1.5% Loss</td>
<td></td>
</tr>
</tbody>
</table>

Table 4-4: List of use cases of Test 2.
4.3.2 Results

All the speech files prepared for the test were subject to the transcription system. And it was tested as a standalone system. Each table of the transcription test result contains reference transcription, hypothesized word sequence, substitutions, insertions, word error rate, and misrecognized words. WER was calculated for every use case and is the standard for performance evaluation. The lesser the WER, the better the performance. The average WER calculated for every use case is given in the following Figure 4-3.

![Figure 4-3: Performance of the configured ASR in test 2.](image)

4.3.2.1 Case 1 and 2

Here it is seen that the WER is minimum and the transcription system performs at its best for these speech files. In case 1 the speech files are 8 kHz up-sampled, while in case 2 the speech files are 48 kHz down-sampled. The speech files are clean with a sampling rate of 16 kHz, so the test data matches the training data of acoustic model.
4.3.2.2 Case 3 and 4

When the noise is added to the clean speech, the performance decreases and a fair increase in the WER is noticed. Both of the samples in these cases were 8 kHz upsampled which is not an ideal 16 kHz sample. As speech files in case 3 were corrupted by higher noise of -9 db as that of -18 db in case 4, so the WER for case 3 was greater.

4.3.2.3 Case 5 and 6

As seen from Figure 4-3, WER of 97% in case 5 is the worst in this test. And there is hardly any transcription done. It has two reasons. The 8 kHz up-sampled speech files were not as ideal as 16 kHz speech files. And the background noise in this case (food court noise) is a non-linear noise with highly varying characteristic. Food court noise in this case contains noise traces of broad-band noise (people talking), and impulsive noise (dropping of items, shouting). The spectral analysis of the food court noise showed clearly the impulses present in the noise which are the worst to handle and corrupt the speech to a greater extent. Moreover the noise level in case 5 was high i.e. -9db. It should be noticed that Sphinx-4 does not have any denoiser, so it performance dies down in the presence of noise. In case 6 the WER is decreased because the noise level of -18 db was lower than that of case 5.

4.3.2.4 Case 7 and 8

The 48 kHz down-sampled speech files were relatively better than 16 kHz speech files. So the WER is lesser than that of case 3, 4. WER is less in case 8 because of the lower noise level.

4.3.2.5 Case 9 and 10

The WER is again very high in case 9 even though the speech sample used in this case is relatively better than 8 kHz up-sampled. It is because of the food court noise and its high level. A decrease in WER for a reduced noise level in case 10 is seen.

4.3.2.6 Case 11 and 12

The speech files were narrow band Speex encoded. In case 11 the performance is better because of the noise source. The WER in case 12 is greater than case 11 even the noise level is lower.

4.3.2.7 Case 13 and 14

The speech files used in these cases were broadband Speex encoded. WER in case 13 is lower due to the noise source. It is also seen that the performance is slightly
increased in case 14 when compared to that in case 9. It is because Speex is a better encoder.

### 4.3.2.8 Case 15 and 16

The speech files in these cases were wideband Speex encoded along with the Speex denoiser implementation. A slight decrease in the WER as compared to case 11, 12 is observed, where the Speex denoiser was not used.

### 4.3.2.9 Case 17 and 18

The speech files in these cases were wideband Speex encoded along with the Speex denoiser implementation. A decrease in the WER is noticed as compared to case 13, 14 where the Speex denoiser was not used. There is a fair decrease in the WER in case 17 where the noise is highway noise at a lower -18 db level.

### 4.3.2.10 Case 19 and 20 and 21

When the noise free speech signal in case 19 is encoded with AMR wideband, the WER is slightly increased as compared to that in case 2. But still there is a fair recognition. In order to evaluate the performance of transcription system in the presence of channel errors delay profiles were added to the noise free speech signals. In case 20 a delay profile of 1% loss is added and the WER slightly increases. The WER further increases in case 21 when a delay profile of 1.5% loss is added.

### 4.4 Test 3

When voicemail transcription functionality was added to Red5 phone solution and the new system was functional, a test was performed to see how the voicemail transcription works in practical. There were three speakers from EAB/TVK. Two of the colleagues had Swedish accent, while one had Asian (Pakistani) accent. Three speech files with different sentences were subjected to the test. The average WER in this test was obtained 43.25 % which is quite satisfactory with this configuration. This performance will increase and higher transcription rate will be obtained once the system is configured with an acoustic model that will be trained on real 8 kHz voicemail speech data.

### 4.5 Speech enhancements results

As it is seen from test 2 results Sphinx-4 performs badly in the presence of noise, and in some cases where the noise level is high the performance totally collapse. So speech enhancement technique must be used in order to reduce the effects of noise. The main idea of every noise suppression method is to obtain the optimal estimate \( \hat{x}(t) \) of original signal from the noise corrupted signal \( y(t) = x(t) + n(t) \). When noise types and conditions vary significantly within the environment in which recognition...
is to be performed, and the environment is open to new and unknown noise conditions, no single noise compensation algorithm can be expected to always perform effectively [16].

Two different noise suppression methods were tested in Matlab. They were noise suppression using Spectral enhancement and noise suppression using Wiener filter. A wiener filter of the method proposed by Plapous was tested [17]. The signals those were obtained after enhancement of noisy speech signals were subject to transcription system. And the results obtained are given in the Figure 4-4. The WER decreases by 8.8 % units when the noise was removed using spectral enhancement. Whereas a fair decrease of 13.75 % units was seen in case of speech enhancement using Wiener filter. So it is recommended to enhance the speech with a Wiener filter prior to subject it to the transcription system. A better approach will be adding an implementation of Wiener filter in Java to the front end of Sphinx-4, as the front end of Sphinx-4 is flexible and pluggable. Or the wiener filter in java can also be added to Sphin4 as a separate modular unit prior to Sphinx-4 front end. So in that case an enhanced speech will be passed to the Sphinx-4 front end.

Figure 4-4: Results obtained from speech enhancement methods.
CHAPTER 5  Conclusion and future work

This chapter concludes the thesis work, and presents the possible continuation of this thesis.

5.1 Conclusions

With the help of Flash enabled browser telephony services can be integrated into it. New services can be introduced that are interesting and useful.

- In this thesis work a prototype of voicemail transcription system was successfully developed.

- It is shown that Sphinx-4 can successfully be used for the transcription of voicemails in a Red5Phone based system.

- The test results showed that the Sphinx-4 is susceptible to background noise and its performance degrades to a fair amount in the presence of it. And in case of a high and severe background noise its performance collapse totally.

- It was noted that the impact of background noise on WER was greater than the impact of a reasonable packet loss.

- Noise suppression is very important and from the results it was seen that, comparatively Weiner filter suppresses noise better. And the WER was decreased to 15 percent units by using a Weiner filter.

- Given an acoustic model trained on real voicemail speech data sampled at 8 kHz, the performance of the introduced service will increase. Better transcription can be achieved by using a proper acoustic model and the WER can be decreased to 16% ~ 18% in a noise free environment. It is claimed on the basis that the only discrepancy in the service is improper acoustic model. With a proper acoustic model the service will give results in the same range as those obtained by using existing 16 kHz acoustic model.

- The newly released Adobe Flash 10 supports the Speex codec. It was successfully tested that better results are achieved with Speex encoding.

5.2 Future work

- Sphinx-4 is written in java and it does not have any noise suppression module. It is very important to have one to improve its performance in the presence of noise. It will be interesting to implement the recommended Wiener filter implementation in java. It can be easily added to the Front end block of Sphinx-4.
• The existing inbox has options of playing, reading, and deleting voicemails. It will be exciting to extend the existing inbox to an advance level. Speech to text synchronization in the advanced inbox will be interesting where the user can click on any word to start listening to the voice message from that word. Other features like searching inbox, forwarding the transcription to emails and as a text message, and setting callback reminders on messages can be included in the advanced inbox.

• When this prototype was developed, issues like scalability and security were not addressed. It is important to have a detailed study of them.

• The NellyMoser codec in Red5Phone can be replaced with Speex (wideband).
References


Appendix A

List of softwares

Here in Table A 1, is the list of technologies required in this service.

<table>
<thead>
<tr>
<th>Software</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adobe Flash</td>
<td>9</td>
</tr>
<tr>
<td>Red5</td>
<td>0.7</td>
</tr>
<tr>
<td>Mini SIP Server</td>
<td>2.9</td>
</tr>
<tr>
<td>MySQL</td>
<td>5.0</td>
</tr>
<tr>
<td>X-Lite</td>
<td>3.0 build 41150</td>
</tr>
<tr>
<td>Sphinx-4</td>
<td>1.0 beta 2</td>
</tr>
<tr>
<td>JDK 6</td>
<td>16</td>
</tr>
<tr>
<td>Ant</td>
<td>1.7.1</td>
</tr>
<tr>
<td>Perl</td>
<td>5.8.8</td>
</tr>
<tr>
<td>Sox</td>
<td>14.3.0</td>
</tr>
</tbody>
</table>

Table A 1: List of softwares.
A portion of the transcriptions is given in the following Table B 1.

<table>
<thead>
<tr>
<th>Speaker</th>
<th>Wave file</th>
<th>Reference Transcription</th>
<th>Hypothesized word Sequence</th>
<th>S</th>
<th>D</th>
<th>I</th>
<th>WER</th>
<th>Misrecognized word</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>a0415</td>
<td>It lasted as a deterrent for two days</td>
<td>it lasted as a deterrent for two decades</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>12.5%</td>
<td>days~decades</td>
</tr>
<tr>
<td>A</td>
<td>a0416</td>
<td>The added weight had a velocity of fifteen miles per hour</td>
<td>he added weight had a velocity of the fifteen miles per hour</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>18.18%</td>
<td>the~he</td>
</tr>
<tr>
<td>A</td>
<td>a0417</td>
<td>It is also an insidious, deceitful sun</td>
<td>it is also an insidious deceitful saud</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>14%</td>
<td>sun~saud</td>
</tr>
<tr>
<td>B</td>
<td>b0026</td>
<td>You have associated with some of these men</td>
<td>you have associated with some of these mean</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>12.5%</td>
<td>men~mean</td>
</tr>
<tr>
<td>B</td>
<td>b0027</td>
<td>And there's no chivalry, no quarter shown in this fight</td>
<td>and there's no chivalry no course are shown in this fight</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>20%</td>
<td>quarter~course</td>
</tr>
<tr>
<td>B</td>
<td>b0028</td>
<td>Lord Fitzhugh is the key to the whole situation</td>
<td>mowed Fitzhugh is the key to the whole situation</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>11.12%</td>
<td>lord~mowed</td>
</tr>
<tr>
<td>C</td>
<td>a0479</td>
<td>I tried to read George Moore last night, and was dreadfully bored</td>
<td>I tried to read George for last night and was dreadfully border</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>16.67%</td>
<td>moore~for</td>
</tr>
<tr>
<td>C</td>
<td>a0480</td>
<td>Tom Spink has a harpoon</td>
<td>tom stink has a harpoon</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>20%</td>
<td>spink~stink</td>
</tr>
<tr>
<td>C</td>
<td>a0481</td>
<td>Nimrod</td>
<td>nimrod</td>
<td>1</td>
<td>2</td>
<td>25%</td>
<td>sensitivness~sensitiv</td>
<td></td>
</tr>
<tr>
<td>replied, with a slight manifestation of sensitiveness</td>
<td>replied with a slight manifestation of sensitive in this</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table B 1: A portion of transcription results.