2010

Phoneme-based Video Indexing Using Phonetic Disparity Search

Carlos Leon Barth
University of Central Florida

PHONEME-BASED VIDEO INDEXING
USING PHONETIC DISPARITY SEARCH

by

CARLOS LEON-BARTH
B. S. University of Florida, 1993
M. S. University of Central Florida, 1998

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the Department of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Fall Term
2010

Major Professor: Ronald F. DeMara

ABSTRACT

This dissertation presents and evaluates a method to the video indexing problem by investigating a categorization method that transcribes audio content through *Automatic Speech Recognition (ASR)* combined with *Dynamic Contextualization (DC)*, Phonetic *Disparity Search* (PDS) and Metaphone indexation. The suggested approach applies genome pattern matching algorithms with computational summarization to build a database infrastructure that provides an indexed summary of the original audio content. PDS complements the contextual phoneme indexing approach by optimizing topic seek performance and accuracy in large video content structures. A prototype was established to translate news broadcast video into text and phonemes automatically by using ASR utterance conversions. Each phonetic utterance extraction was then categorized, converted to Metaphones, and stored in a repository with contextual topical information attached and indexed for posterior search analysis. Following the original design strategy, a custom parallel interface was built to measure the capabilities of dissimilar phonetic queries and provide an interface for result analysis. The postulated solution provides evidence of a superior topic matching when compared to traditional word and phoneme search methods. Experimental results demonstrate that PDS can be 3.7% better than the same phoneme query, Metaphone search proved to be 154.6% better than the same phoneme seek and 68.1 % better than the equivalent word search.

Dedicated to my Wife, Mom & Dad.

ACKNOWLEDGMENTS

First and foremost, I want to show gratitude to my family and closest friends for their never-ending support of all of my endeavors. I would like to thank Drs. Ronald DeMara, Avelino Gonzalez, Eduardo Divo, and Shaojie Zhang, for serving on my dissertation committee. Much appreciation goes out to my Intelligent Systems Laboratory colleagues – Miguel, Victor, Raul, J. R., and Rueben. I would also like to thank L3 Communications Internal Research and Development, Robert Williams at UCF, and Don Harper at UCF EECS Computer Support for their economic support that make this important research a reality. A show of gratitude goes out to Alex Schwarzkopf and Rita Rodriguez at the National Science Foundation for their continued financial and moral support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS/ABBREVIATIONS

ATWV	Actual Term Weighted Value
AC	Aho-Corasic Pattern Matching Algorithm
AI	Artificial Intelligence
AIML	Artificial Intelligence Markup Language
API	Application Programmer Interface
ASR	Automatic Speech Recognition
CMU	Carnegie Mellon University
DB	Database
DM	Dialog Management
DNA	Deoxyribonucleic Acid
DOD	Department of Defense
FSM	Finite State Machine
GAUDI	Google Audio Indexing Technology
GUI	Graphical User Interface
GUID	Global Unique Identifier
HCI	Human-Computer Interaction
I/UCRC	Industry and University Cooperative Research Program
IE	Information Extraction
IR	Information Retrieval
ISI	Intelligence and Security Informatics

ISR	Isolated Speech Recognition
IV	In Vocabulary Words
LSI	Linear Spline Interpolation
LVCSR	Large Vocabulary Continuous Speech Recognition
MED	Minimum Edit Distance
MMR	Maximal Marginal Relevance
PDS	Phonetic Disparity Search
QUT	Queensland University of Technology
SAPI	Microsoft Speech API
SCTK	NIST Scoring Toolkit
SLG	Spoken Language Generation
SLU	Spoken Language Understanding
SMS	Short Message Service
SMT	Statistical Machine Translation
STD	Spoken Term Detection
TREC	Text Retrieval Conference
TTS	Text To Speech
TWV	Term Weighted Values
NLP	Natural Language Processing
NSF	National Science Foundation
NIST	National Institute of Standards and Technology

UCF	University of Central Florida
WAV	Waveform Audio File Format
WER	Word-Error Rate
WCN	Word Confusion Network
WS	Word Spotting
WWW	World Wide Web

CHAPTER ONE: NEED FOR PHONETIC SEARCH METHODS

Video and audio categorization, supporting audio search and retrieval, evolves as a recent research topic as increasingly large media libraries become progressively more difficult to explore. These massive databases present a data mining challenge, as attempts to index these collections have proven ineffective due to the acoustic nature of the content. Furthermore, the evolution and fusion of Context Based Summarization and *Automatic Speech Recognition (ASR)* empowers researchers with the ability to translate audio documents at the expense of imperfect recognition accuracy, a metric known as *Word Error Rate (WER)*. However, scholars and industry recently debate if the WER measure is the paramount measure of speech recognition error (Wang, Acero, & Chelba, 2003). Nevertheless, we demonstrate that Indexing and searching audio and is possible regardless of the WER by extracting phoneme, word, and sentence information from the audio content. Furthermore, we demonstrate that it is possible to optimize video storage using a blend of summarization, phoneme conversions, and genome pattern matching search algorithms mixed with phonetic disparity search and Double Metaphones (Preisach et al., 2008). We believe that the out of vocabulary (OOV) words induced by ASR translation errors can be found using phonetic methods since the sound of each word in fairly preserved in the phonetic conversion of the utterance.

Challenges Facing Automated Speech Recognition and Indexing

Speech recognition systems are generally classified or as discrete or continuous systems that are speaker dependent or independent. Discrete systems keep a separate acoustic model for each word combination that is sometimes referred as Isolated Speech Recognition (ISR) systems. Continuous speech recognition (CSR) systems respond to a user who pronounces words or phrases that are dependent with each other, as if they were linked together. The speaker dependent system requires a user to train the system; thus, each spoken word must have an equivalent sound in the subset system vocabulary to be matched. Speaker-independent systems do not require to record voice prior to the system and work with any type of English speaker. The third and more novel speech recognition system is the speaker adaptive system; it is developed adapts its operation to the characteristics of new speakers. Most modern speech recognition systems use probabilistic models to interpret and compare the input sounds with an internal library of sounds known as the ontology. The Ontology is where two grammars subsist, one for the speech recognition process and the other for the speech generation as in IBM VIA VOICE (Bianchi & Poggi, 2004).

Grammars contain what the ASR system knows about the input sounds that it receives. Modern ASR systems provide automatic training of the contextual side of the equation, and do provide a standard audio ontology that works with the average user. Commercial ASR engines provide grammars based on a context where the speech recognizer will interact with a human. However, specific applications that are catered to support and specific application or scenario require customized grammars to support the context of the conversation. Previous research that addressed the interaction of an avatar with humans in a kiosk like fashion provided successful support to a topic due to its knowledge of the context of the application. The project LIFELIKE, was successful in creating an avatar that served as a surrogate of the NSF director. The kiosk like implementation provided an interactive speech user interface that provided information about the I/UCRC NSF program. LIFELIKE at its initial stage, used grammars to support the questions asked about the program, however the manually created grammars did not provide the LIFELIKE concept of natural speech. Further research, moved from customized grammar sets to automatic document training using Windows 7 ASR. The ASR ontology was then trained using documents that contained the vocabulary used to support I/UCRC. For the audio training, and average male voice was used to read the same documents to provide further audio ontology training. Using the ASR dictation mode, LIFELIKE was able to provide an increased natural response, with the support of a dialog manager that constantly searches for key words that match the trained ontology within context (DeMara et al., 2008). However, the result to provide accurate responses is limited due to ASR imperfections. It is well known that ASR context based training significantly improves the WER (Word Error Rate), however the best well trained Large Vocabulary Continuous Speech Recognition (LVCSR) are no better than 24.8 % according to TREC-7 results (Johnson, Jourlin, G.L, Jones, & Woodland, 1998). A year later on TREC-8, using a 2-pass HTK speech recognizer which ran at 15 times real time, scored a word error rate of 20.5% using a 10 hours subset of the original 500 hour corpus (Johnson, Jourlinz, Jonesz, & Woodlandy, 1999).

The size of the vocabulary is directly proportional to the word error rate. Larger vocabularies can generate similar sounds for different words making it harder for the speech recognition system to match the correct word. Discrete speech recognition systems require pauses between the words. Continuous speech recognition systems analyze an utterance at a time; therefore, are recognizing a group of words at a time. Hidden Markov Models and Verturbi algorithms are probabilistic tools used to find the most probable next hidden word based on the previous words (Obermaisser et al., 2007). Most continuous ASR's on the market today use HMM to provide speech recognition, to mention a few IBM VIA VOICE, NUANCE Dragon and Windows 7 SAPI. The theory behind the creation of speech recognizers and speech synthesizers is beyond the scope of this work, but are mentioned for reference. The above-mentioned ASR's are speaker independent; however work well with an average male voice, but require previous training.

Phonemes are a group of different sounds that individually, are the smallest segmental unit of sound needed to compose meaningful thought. A phone is an individual sound considered a physical event regardless of it place. Therefore, a group of phonemes compose a word phonetically speaking, and multiple phonemes form an utterance. An Utterance is a complete unit of speech in spoken language; it may not by separated by silence ("Phoneme," 2010).

In general, speech recognition systems processes input voice data through a recognizer that matches the input with an acoustic model that through decoding, generates a hypotheses of the most probable sequence of words that match the original voice input. At the same time, it is possible obtain a sequence of phonemes characterized by individual phones from the same hypothesis. We believe that phonemes better describes the sound of the input sequence or utterance as it mitigates the errors caused by the ASR translation or Out of Vocabulary words (OOV). We consider that, OOV errors can be diminished due to ASR errors in translation, by using phonemes, since the semantic sound is preserved somewhat in the original word. As the phonemes are generated, it is possible to store their phonetic utterance without any modification four indexing and posterior search. Further categorization can be made through algorithms that searched the phonetic stream, and provide algorithmic conversions for further analysis and indexation. At this stage, the phonetic information is categorized for phonetic search employing different methods. Figure 1, illustrates an utterance captured from a Microsoft Windows 7 ASR output. Notice that there is no separation between the phones. I is indistinguishable were each word starts and ends. Nevertheless, it describes the utterance phonetically.

Manual transcript: Members of the Harvard Corporation.

ASR translation: members of the harvard corporation

ASR Phonetic Utterance: m eh m b ax z ax v dh ax h aa r v ax d k ao r p ax ey sh ax n

Figure 1: ASR and Phonetic outputs for a Windows 7 ASR

It is apparent that the phonemes represent the sounds of the original transcript. This dissertation provides proof of a methodology that regardless of WER, ASR conversations can be indexed and searched through phonetic strings or utterance conversions captured by the ASR and later categorized and stored a relational database for phonetic search.

Other factors that deteriorate speech recognition are environmental factors, i.e. external noise, multiple speakers from different directions, the source of the recorded audio. It is know that recordings made with unidirectional microphones fair better than conference room recordings made with a central microphone. Other environmental factors are the acoustics of the room, second and third arrival sounds, and clipping of the audio input where the source is so loud that causes distortion on the recording further diminishing the quality of the input signal. On video recordings, we mentioned that clapping, commercials and any other signal that is not the voice of the speaker significantly diminishes the quality of the audio signal.

Research efforts continue to clean the input signal before it is converted. Recent Microsoft Research article proposes a Linear Spline Interpolation (LSI) to predict noise. They demonstrate that there is a non linear relationship between clean and noisy speech, that LSI can update the noise channel unsupervised with improvements on 10.8% when compared with their own baseline (Seltzer, Acero, & Kalgaonkar).

Demand for Phonetic Speech translation and indexing

The demand for categorization and search of information is not new; however as media indexation is added into the equation, traditional character categorization and search alternatives prove inefficient (Makhoul et al., 2000). Database technology does not yet provide automatic indexing methods for media extracts except for available storage as blobs; however, the indexation of the content chunks or utterances is left to the designer. The development of new theory and algorithms is needed to outsmart the long-established data mining and categorization schemes used for media conversions such as video, audio, pictures and animation. Interestingly, flourishing solutions to these issues are not only due to technology enhancements, they are result of technology integration imported from disparate areas, or computational implementations of efficient but unrealized manual methods. (e.g. *Soundex,* a phonetic categorization method for names used by the U.S. Government Census and analyzed on previous research in Australia (Justin & Philip, 1996). Therefore, the relevance of other sciences in the abstraction of media content becomes a technical challenge regardless of the content of the audio. Phonetic transformations of the conversational audio and video look promising in speech mining, but prove useless in abstract video and audio content such as instrumental music. Increasingly difficult is the categorization of music over other recorded media, because it does not convert into a searchable medium due to the lack of speech. Research in areas of signal analysis and digital signal processing, propose wavelet pyramidal algorithms (Ying & Yibin, 2004) and Audio Finger Printing (Ling, Yaohua, Yun, & Yong, 2009) within others, however beyond the scope of this document.

Business and Military Intelligence is searching for ways to interpret large amount of telephone or similar audio conversations (Reddy, 1976). Structured and unstructured audio data provide access to businesses information dynamically of the customers call center data, email and SMS, and more importantly information about trends that are difficult to derive otherwise from a large population of customers (Subramaniam, Faruquie, Ikbal, Godbole, & Mohania, 2009). Intelligence and Security Informatics (ISI) discipline provides research ideas in areas of Data Mining of phone conversation that provide insight to the use of categorization of knowledge that addressing National Security issues (Hsinchun & Fei-Yue, 2005). By collecting audio information from select telephone conversations, relevant information could be categorized and searched for Intelligence.

Business intelligence research has publicized the need to capture open dialog from call center conversations, and categorize them for market research as the industry minimizes the operating cost. “Typically, call center human agents cost US$2–$15 per call, while automated dialog systems cost less than US$0.20 per call and are 40% faster on average (Gilbert, Wilpon, Stern, & Di Fabbrizio, 2005). It is observed that modern studies continue to struggle with analysis of recorded data efficiently and gainfully, while the automatic information categorization and intelligent analysis of the data prevails as an elevated priority. Improvements to Automatic Speech Recognition (ASR), Spoken Language Understanding (SLU), and Dialog Management (DM), as well as Spoken Language Generation (SLG), and Text-to-Speech (TTS) synthesis, allow the intelligent categorization and summarization of speech within the technology limits, nevertheless proven practical. SLU experiments conducted on a semantic performance using an AT&T engine with call center data in the context of healthcare and telecom with 100% semantic relevance, proved no better than 65% when Semantic Precision versus Semantic Recall was compared. Less precision increments the recall, but gives less accuracy, a symptom observed in our results with PDS. For High precision, the Recall was less than 40%. No information was posted regarding the phonetic translations in the comparison, even though ASR translations were used within the process. Call center data call quality is analyzed using ASR conversions and data is used for business Intelligence (Zweig et al., 2006).

Experimental Summarization Algorithms are widely available; they provide topic information about documents by further augmenting the original content with web-based searches. Then, if media can be converted into, words phonetic are summarized at the current ASR WER. Novel methods evolve from the research community that summarizes social networking by analyzing the data contained in tags and comments made by the users of social networking sites (Jaehui, Tomohiro, Ikki, Hideaki, & Sang-goo, 2008). However, the indexation of the results using a phonetic approach for accurate seeks was not explored.

Numerous methods are currently been explored to analyze, categorize and store video, text extraction through ASR leads the way. The National Institute of Standards and Technology (NIST) studied the effectiveness of speech recognition in spoken documents on their Text Retrieval conference TREC track sponsored research. Spoken Term Detection (STD) is a recent term employed by NIST that describes the search of specific terms on ASR translated content. The challenge presented was to locate occurrences of a specific list of words in a given corpus on broadcast news, telephone conversations, or recorded meetings using ASR translation. The classic search method for a set of words is to translate large audio content using a content-trained ASR. As the translation is obtained, the output is indexed and stored for posterior search. The subsequent search is based on queries that aim at a particular set of words. Word matches are further analyzed and indexed for pier comparisons. However, a search *for Out of Vocabulary (OOV)* words of similar context will not return direct results since these words were never part of the translated ASR document. Further mismatching category can occur if the speech recognizer's miss-translated words are considered, while similar in phonetic construction, inherit a disjoint semantic relationship to the original word, therefore not found by traditional corpus searches. Active research, attempts to solve the above mentioned issues with ASR translations and further reconstruct the translated document by decreasing the mismatch of words due to inherit ASR errors. The goal was to measure the performance of multiple ASR engines converting standardized broadcast dialog as the main corpus. No experiments were found that conducted research using a phonetic approach.

Today, speech recognition research is interdisciplinary, where fields such as biology, computer science, electrical engineering, linguistics, mathematics, physics, and psychology intertwine in areas of acoustics, artificial intelligence, computer algorithms, information theory, linear algebra, linear system theory, pattern recognition, phonetics, physiology, probability theory, signal processing, and syntactic theory. The phonetic indexing and searching of video and audio content is one of the many applications of categorizing ASR translation data. This dissertation work addresses methodology imported from the phonetics, pattern recognition, artificial intelligence, computer algorithms and speech recognition to provide another possible solution and application to the audio search problem through phonetic indexing.

Performance Metrics

Establishing guidelines and promoting research in this area is the National Institute of Standards and Technology (NIST). Within other standards, NIST established metrics for ASR performance. Historically, NIST has evaluated and conducted ASR engine performance measuring WER by converting newscast media, speeches or spoken dialogs from close to ideal environments. Content transcriptions were made available to selected research teams to generate comparison WER estimates using different ASR engines but identical media. Although initial NIST results were published in 1999, the methodologies that flourished from the research are prevalent for WER calculations today (Johnson et al., 1999). At present, NIST provides a Speech Recognition Scoring Toolkit to estimate Word Error Rates (NIST, 2010). Similarly, Carnegie Mellon University provides a similar alignment tool “Align” that does not provide all the functionality of the NIST counterpart, but allows calculating WER rates using Microsoft SAPI ASR output. Recent NIST’s Spoken Term Detection evaluation augmented new research that suggests a different approach and metrics. However, this NIST sponsored research has been neglected since Dec 2006. Scheduled revival of such research in 2008 did not flourish.

The calculation or WER requires an alignment of a manual transcription of the audio or media and the ASR translated text output. It is a known fact that the output will not align with the original manual transcript due to the insertions, deletions, and substitutions caused by the ASR errors. However, as the ASR converts the audio signal into text, and it does it one utterance at the time. For calculation of the WER, the alignment of each utterance produced by the ASR has to be aligned with the corresponding section of the manual transcript. This is a tedious process even if a transcript of the original content is provided. TREC research provided not only the audio content of hours of recorded audio, but also the transcription of such content. I the particular case of this dissertation, we selected video recordings of news panel discussions regarding politics and the economy. Despite the fact that we had access to transcribed material, it was not aligned properly with the utterances provided by Windows 7 SAPI ASR. Consequently, utterance by utterance was saved separately by the SAPI call back algorithm and manually aligned with the original translation of the audio. As a remark for the reader interested in duplicating this process, the alignment of the ASR translation is not a straightforward process specifically in cases where the speaker emits disrupted sounds by involuntary repetitions and prolongations of sounds, syllables, words, or phrases. The addition of involuntary silent pauses due to the speaker inability to produce sounds causes a misalignment of utterances with the original transcription. The key is to align beginning and end of an utterance with the original transcription. A daunting task when the edges of the utterance present uncertain results not considered in the original translation, but necessary for alignment. The deletions, insertions and substitutions caused by misalignment will be reflected in the WER error calculations.

Current Commercial-off-the-shelf (COTS) speech recognition systems strive to provide the most probable text output to an acoustic input signal. In the process, WER is greatly affected by the speech recognition engine used, the different language models and the data used for pre-training. Consequently, ASR transformations are far from accurate. As postulated by NIST, WER is defined in the next equation (1). NIST further defines hypothesis as the best possible translation generated by the ASR from recorded audio. To calculate WER, an alignment between the reference transcription and the hypothesis is first necessary. Then an estimation of the number of word Insertions (I), Deletions (D) and Substitutions (S) is calculated and divided by the original word count of the reference document. Notice that our experiments are conducted using a CMU's version of NIST's SCTK tools to estimate WER results. Our tests are performed using Windows 7 SAPI that misses providing callback data for alignment, information that NIST experimental ASR's provide. Indeed, CMU has made software available that permits the alignment of both text strings and simplifies the process; however pre-processing the data is necessary but simplified. Perhaps, no experiments were conducted at NIST using Microsoft SAPI.

$$WER = \frac{(S+I+D)}{W} x\ 100 \tag{1}$$

NIST sponsored speech recognition track Text Retrieval Conference (TREC) studied speech recognition performance in the late 1990's, concluding that the accuracy of multiple recognition engines was at best no better than 24.8% WER when using Cambridge HTK ASR (Johnson et al., 1999). By the end of the year 2000 the Spoken Document Retrieval was considered solved (Garofolo, Auzanne, & Voorhees, 2000). However, the ASR transcribed documents reveal weaknesses while performing OOV word searches. Perhaps, no results return from direct searches, a reason to hypothesize that the phonetic information regarding the ASR translated text, was omitted as part the study. Recent work related to ASR performance compared SAPI (Microsoft, 2009) against SPHINX4 (CMU, 2008) in a framework that was developed to control robots through speech. On this study recognition rates were on average 84% lead by SAPI's recognition rate of 98% using JDK5.1 and Windows 2000 (Ayres & Nolan, 2006). Disappointingly, no information is given in how the Recognition Rate is calculated, therefore difficult to compare with WER. However, if Recognition Rate it is considered the opposite of WER a Recognition Rate of 84% infers a 16% WER, a score better than any NIST TREC results. Perhaps, results only verified under ideal conditions and a well trained ontology.

No others studies, appear to be available, that provide information about the WER of Windows 7 SAPI as used on our experiments.

Regarding the use of Databases (DB) for general storage and organization, the retrieval process theory itself postulates metrics to evaluate the relevancy of a search or query. Queries are formal statements of information needs. Queries do not identify a single object in the collection, on the contrary queries my return several objects within the collection with a certain degree of relevancy. Moreover, objects can be defined as the items of information stored in a database. Depending on the application a query a match text documents, images or videos, however not until recently, the files are not stored directly inside the DB. Instead, they are stored as surrogates' metadata that describe the content or location of the original file. Different measures are used to quantify the performance or the Information Retrieval (IR) system based on the relevancy of the query. Within a single query, they may be different forms of relevancy. With the large text collections available from TREC, old methods were modified and new techniques evolved for effective retrieval of large documents. TREC branched in other IR fields such as retrieval of spoken information, non-English language retrieval, information filtering, user interaction with IR within others (Singhal, 2001). The Information Retrieval Theory is extensive and continuously being redefined. Within the scope of our research we will define Precision, Recall and Fall-Out; as well as F-measure, Mean Average Precision and Discounted Cumulative Gain ("Information Retrieval," 2010).

The Precision metric is the proportion of objects retrieved that are relevant to the user. Precision considers all the matched documents.

$$Precision = \frac{|\{relevant\ docments\} \cap \{retrieved\ documents\}}{|\{retrieved\ documents\}|} \quad (2)$$

The subsequent metric used is Recall and is defined as the proportion of objects that are relevant to the query that were successfully retrieved.

$$Recall = \frac{|\{relevant\ docments\} \cap \{retrieved\ documents\}}{|\{relevant\ documents\}|} \quad (3)$$

Fall-Out is the proportion of no relevant objects to the query that were retrieved, out of all the non-relevant documents available.

$$FallOut = \frac{|\{non-relevant\ docments\} \cap \{retrieved\ documents\}}{|\{non-relevant\ documents\}|} \quad (4)$$

F-Measure is the weighted harmonic mean of Precision and Recall, also known as F-Score. The enunciation of database performance equations is listed here as a reference. Performance recall analysis of our results will be presented in chapter six.

$$F = \frac{2 \cdot precision \cdot recall}{(precision + recall)} \quad (5)$$

Approaches and Limitations

Our initial idea was based on the premise of possible video to audio conversion followed by speech recognition using capabilities of Windows 7 operating system. We anticipated stripping the audio from the video content and converting it into phonemes and text, using the available ASR engine. Then, the text and phonemes from each utterance conversion could be analyzed and categorized as we stored all output in a relational database. We further augment the retrieved content by adding conversational topical information, using available summarization techniques. Genome pattern matching techniques were also explored and used to index the available translations of the original video content. A separate search interface would serve as a benchmark tool to test the different phonetic search algorithms implemented and their capabilities verified.

Every stage of the process opened new possibilities, but also presented its own limitations. The conversion of the video into the audio presented its own caveats. The video material contained embedded advertising from the original TV recording. Consequently, each hour of the News broadcast video had to be scrutinized for clutter not related to the normal dialogue of the broadcast. In addition, the video source web site occasionally interrupted the playback as we recorded news content voiding the recorded sample. Multiple takes were necessary to accomplish a clean video source. The cleanup process mired the ability to automatically convert video to audio. Similar studies use a waveform transcoder which extracts the audio signal from the videos and down sample it to 16 kHz further filtering noise or clutter out of the original signal, losing some content in the process (Alberti et al., 2009). In our approach, we only cleaned the original video signal from commercials; however, ambient noise or crosstalk was not subtracted from the original content regardless of noise. Our intent was to keep aligned, as much as possible, the audio and the original video; reason for which we filtered the video and then striped the audio that was later used for ASR conversions. We then processed 5 hours of video into audio and stored it in a repository indexed by the DB.

For the audio to video conversion we used Gold Wave ("GW," 2010), and converted the incoming audio signal at a sampling rate of 16 kHz,16 bit mono channel. Higher sampling rates are available but most of the Speech Recognition research uses 16 kHz at 16 bits mono audio conversion.

Using the available windows 7 SAPI the audio (WAV) file was loaded into the ASR for text and phoneme extraction. Each utterance was captured separately as converted by the ASR. The original audio content was transcribed and aligned with the ASR text output for WER estimation. During the alignment process, we notice a need to train the Speech Recognizer since the initial output was illegible. Although we noticed a fair translation of the original document, some of the OOV words inserted were evidence of improper recognition, even on utterances that contained very clear dialog. Thence, further training was provided by feeding the system with documents with related topics, for Windows 7 SAPI to automatically train its own vocabulary by analyzing the documents stored under a specific path and providing itself with additional language.

ASR speech recognition results can vary; they are directly affected by the trained corpus, speaker number and diction, and the environment. Single user speech recognitions systems fail to recognize different voices, and can become unstable when background noise is present. The best results are achieved with systems that use a Large Vocabulary Continuous Speech Recognition (LVCSR). Under controlled conditions, ASRs can provide accurate transcriptions of selected data yielding 90% recognition accuracy. (Jonathan, Bhuvana, & Olivier, 2007). Nevertheless, most speech dependent recognition engines can achieve high levels of performance when trained, and in controlled conditions; however, automatic speech independent recognizers have limitations. Most important, is the ability to provide the most probable hypothesis based on a trained lexicon about a certain topic. As the lexicon size is increased, the ability to discern from similar utterances becomes more difficult. Indeed, the quality of the audio material significantly affects performance. Multiple speakers, background noise, microphone and room acoustics, all contribute to inaccurate recognition. Furthermore, the diction of the speaker or speakers and the amount of crosstalk in the recorded material also contribute to high WER rates (Shriberg & Cetin, 2006).

The key issue with spoken language processing is the integration of speech and language understanding (Sang-Hwa, Moldovan, & DeMara, 1993). Natural language Processing (NLP) is beyond the topic for this research, however considered a new wave for analyzing the context of a conversation.

As a result, video indexing is a direct application of Spoken Term Detection (STD) and an OOV improvement. The similarity can be found as video content is translated into audio and further translated to text using speech recognition technology, the content is indexed for later detection and mapped to the original video. Indexing and Search technologies can then be used to augment the STD problem, and perhaps provide insight on OOV text reconstruction.

Similarly, we propose a system that extracts audio content from live video and categorizes it. However, our goal is to summarize the content of the video automatically, by presenting not only metadata for automatic indexing of the content, but also delivering time aligned content for user search based on ASR phoneme transcriptions of the audio. Thus, minimizing the error from ASR word translations is possible since phoneme representations while perhaps distorted by the ASR conversion, preserve the original utterance. On the contrary, utterance audio is lost when the most probable word is substituted by the ASR language model, therefore inserting OOV errors.

Multidisciplinary sciences are now looking at the problem from different perspectives and prepare to provide diverse insight to the problem.

CHAPTER TWO: PREVIOUS WORK

This chapter describes the current state of the art techniques associated with Spoken-Term Detection Systems and speech-based phonetic indexation storage and search. The previous chapter identified the metrics, approach, and limitations of phonetic interpretation of speech. Previous results demonstrated a 52% correct phonetic transcription with only 12% inserts using a acoustic-phonetic continuously variable duration hidden Markov model (Zweig et al., 2006).

Recent work focus is reducing the OOV words in LVCSR transcriptions by different methods using different sets of test data. Only NIST, Spoken Term Detection (STD) Evaluation Track, explored the ability to process audio using LVCSR scrutinized by non-traditional metrics using a standardized set of test data and metrics. STD defined by NIST is the ability find word sequences rapidly and accurately in large heterogeneous audio (NIST, 2008). Independent research suggests different approaches to the same problem by using linguistic and stochastic approaches to reduce the OOV words by repairing the ASR translations or performing hybrid searches, while commercial products emerge from these technologies in promises of automatic indexing audio or video.

OOV and Spoken Term Detection Systems

A corporate participant of NIST STD evaluation, IBM Research, suggested a vocabulary independent system, used a blend of phoneme and word translations independent from the vocabulary used. On their approach, the speech recognizer generates confusion networks and phonetic lattices while the transcripts are indexed for querying. Traditional searches of OOV word produce no results because the OOV are missing terms. The suggested approach keeps track of the timestamps for both phoneme and word translations while indexing, to create a merger between translations aligned by time. The OOV scoring was based on the time proximity of the phones in the translation while the scoring for in vocabulary words was based in a Word Confusion Network (WCN) (Jonathan et al., 2007). The research team suggests that phoneme translation and phoneme searches suffer from low accuracy while word-based approaches suffer from an incomplete vocabulary. Therefore, each solution has its faults, but the suggested hybrid solution compares 5% better than phones or words alone.

$$TWV(\theta) = 1 - average_{term}\{P_{Miss}(term, \theta) + \beta \cdot P_{FA}(term, \theta)\} \quad (6)$$

where:

$$\beta = \frac{C}{V} \cdot (Pr_{term}^{-1} - 1)$$

θ= detection threshold

STD NIST Evaluation Workshop provided test results of 10 participants doing about 1000 searches on 10 hours of material. On this track, private companies and academia join their efforts to test Term Weighted Values (TWV) on large audio content. Measurements for the tests were based on a TWV value of one for perfect score, where no values were lost by the system. Calculations for TWV are defined in = detection threshold (Fiscus, Ajot, Garofolo, & Doddington, 2006).

The highest Actual Term Weighted Value (ATWV) for English was 0.83, while the lowest value approached -0.1. Better results were possible using Broadcast Media, followed by phone conversations while meeting data scored the lowest.

Figure 2: English Actual Weighted Term Values represents the ATWV results obtained by the different participants of the workshop. If we consider only Broadcast News, it generated

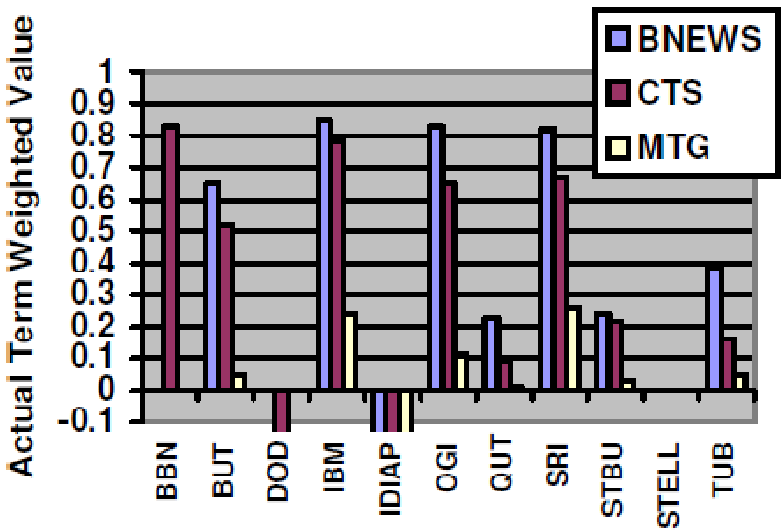

Figure 2: English Actual Weighted Term Values (Fiscus et al., 2006)

the highest Actual Weighted Term Value for all participants with the exception of DOD and IDIAP that had negative results. We believe that broadcast news outcomes the best results because most of broadcast news presents few crosstalk and less background noise. Moreover, News Broadcast record the audio signal directly from the source, each participant has their own microphone diminishing second arrivals from the source. Meeting recordings, on the other hand are typically recorded from a single source located at bets in the center of the room. Such content is more receptive to noise and second and third arrivals, as the sound bounces in different surface areas before reaching the recording microphone. Furthermore, interruptions and crosstalk from all the participants is prevalent, and further distorts the final audio recording as seen in the graph. Phone conversation media, while less prone to crosstalk, has poor bandwidth with poor signal to noise ratio. However, phone conversation material fairs an average ATWV of approximately 0.49 for all participants in the test. That means that only half of the searched values were to be found using all ASR together. Interestingly, Broadcast News alone, regardless of the speech recognition engine used, soared, and average ATWV is about 0.56%. Again, only about half of the terms were found using all ASR engines together. It seems that for any ASR engine used on any study that yields an ATWV higher than 0.5 is performing very well considering the competition. The quality of the audio sample is critical for the performance of the ASR conversion.

Independent research prefers the use of their own test files and transcripts for convenience (Alberti et al., 2009). It is easy to understand why, after waiting for 10 hours to complete an ASR-video translation with a conversion speed ratio of 1:1.

QUT Research performed in Australia Spoken Term Detection research using phoneme extraction (Wallace, Vogt, & Sridharan, 2007). On this research, the search of terms requires the human translation of a baseline document into a phonetic sequence, which is used to find or detect close matching phonetic sequences. This approach provides a fast vocabulary search without the use of a LVCSR engine.

Other commercial systems that use comparable approaches are Virage Audio Logger (www.virage.com), Nexidia's Fast-Talk invention (www.nexidia.com) and Convera's product (www.convera.com) to mention a few. Additional thesis and research before 2004 has been listed by (Saraclar & Sproat, 2004)

Independent research explored the effect of using both word and sub-word information to perform OOV and in-vocabulary searches, particularly showing discrepancies between the search accuracy and the audio transcription speed. The research group found that hybrid systems perform better than fuzzy search (Ramabhadran, Sethy, Mamou, Kingsbury, & Chaudhari, 2009). Results were evaluated using 2006 NIST SDT data.

Searches containing OOV words present a challenge. The search for OOV is impractical since he words have been lost in the translation either replaced by probable similarities or missing. The effects of OOV words is studied by (Woodland, Johnson, Jourlin, & K. Spärck, 2000). The team suggests that OOV error rates decrease sub-linearly with the size of the corpus. The same group concludes that OOV word can be diminished using advanced techniques such as document and query expansion, methods that collect the lowest frequency words from a group of translations and use them to expand the ASR vocabulary.

Note that WER estimation in our experiments is done by comparing the original manual transcripts of a selection of newscast video, and the counterpart ASR recognition text output. The test software aligns both text strings and calculates the WER based on CMU WER script for reference.

Regarding WER studies, it has been demonstrated that ASR WER can be improved (Zechner & Waibel, 2000) by including a human-in-the-loop to provide summaries of the text and merging those computationally with the ASR output using Maximal Marginal Relevance

(MMR) (Carbonell & Goldstein, 1998). Correlated results were positive although varied on each of the four video samples studied. Results proved neutral and improved WER after summarization. Nevertheless, a remarkable improvement when the entire collection was considered. Interestingly, on this research, we consider computational summarization combined with pattern matching techniques to improve WER and, therefore, ASR improved translations.

In this discussion, we use a process of analyzing the contexts of conversations to determine the topics relevant to discussions. We call this process *gisting*, which performs Natural Language Processing (NLP) analysis on textual transcripts through various models to recognized named entities and important phrases.

ASR translations are imperfect in nature, with WER that vary widely between 24% to 66% (Johnson et al., 1999). However, pattern matching has been used by genetic computational research to find different proteins or genes in large DNA sequences. Our study imports these techniques to utilize them to find phonemes within the phoneme based ASR translation regardless of WER accuracy. We postulate that by blending these technologies to build a combined phoneme contextual indexing complemented by a phoneme disparity based search, provides a fast reliable seek regardless of WER and applicable to large media content management.

Pattern Matching Approaches

Pattern matching is not a new science. With the ability to recreate DNA sequences, identifying these sequences within species is a topic of research beyond the scope of this document. Nevertheless, the search of sequences within DNA could not be done without pattern matching algorithms. Quite a few algorithms have grown into intelligent mutations that solve DNA sequence searching. Interestingly enough, phoneme representations accumulated in a string compare favorably to a DNA sequence when it comes to find possible methods to search for patterns. The phoneme strings that are extracted from an ASR translation can be imagined as a finite set of characters with a beginning and an end, but with no distinction between utterances. It is here where ASR translations meet genome pattern matching algorithms.

Pattern matching seeks the occurrence of a particular pattern or characters in a large string of text. Exact pattern matching searches all the occurrences of a pattern of ***m*** characters in a test of ***n*** characters based on a finite alphabet set $\sum$ of size σ

$$m\ (x=x_1, x_2, x_3 \ldots, x_m)$$

$$n\ (y=y_1, y_2, y_3 \ldots, y_n)$$

I general, Pattern-matching algorithms use a window to scan trough the data in search for the pattern within the window known as an *attempt*. Alignment of the window is crucial therefore aligning the left side of the window with the text is first followed by matching the subsequent characters in the window.

As the match fails, the window traverses throughout the entire text in search of a pattern. At this point, the algorithm varies and it becomes specific to the particular research and pattern-matching application. Hundreds of pattern matching algorithms are available that perform better or worse based on the pattern length, periodicity and alphabet size. There is two phases to the pattern-matching, the preprocessing phase and the searching phrase (Thathoo, Virmani, Lakshmi, Balakrishnan, & Sekar, 2006). The preprocessing phase prepares the document and the window to minimize the search effort in phase two. Phase two attempts to find the pattern or patterns minimizing the time of the search and providing maximum efficiency.

Known algorithms that cater to improve the shift value are for example Boyer-Moore (Boyer & Moore, 1977), Quick Search (Sunday, 1990) and Berry-Ranvindran (Berry & Ravindran, 1999) within others. Specifically, we use the Aho-Corasick (Aho & Corasick, 1975) that uses a word tree instead of a traversing window. It can perform a search for multiple patterns at once based on a tree structure where the nodes represent the symbols of the patterns searched.

The Aho-Corasick (AC) better suits our problem because we have large sets of phoneme patterns we need to match within the ASR translation. The suggested algorithm locates all the occurrences of any finite number of keywords within a string of text in a single pass. At runtime,

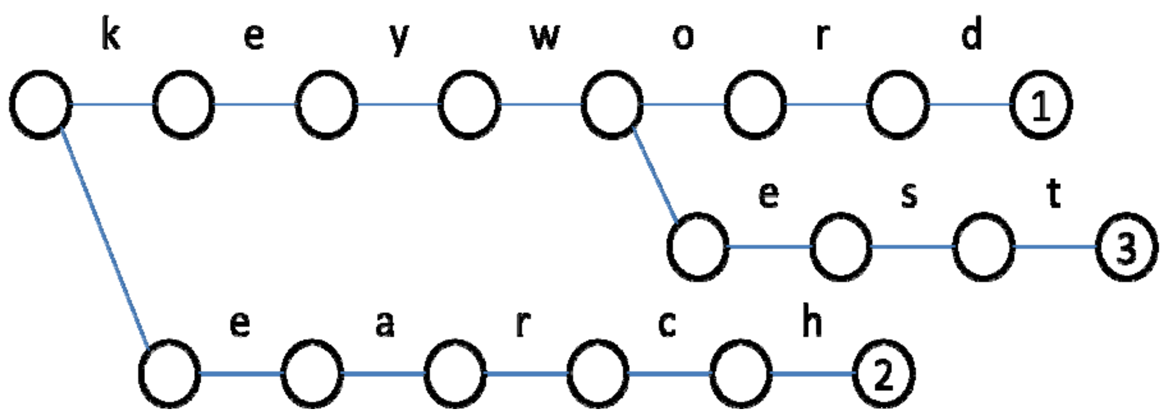

Figure 3: Aho-Corasick Keyword Tree

AC initially creates a tree of words that will be traversed as the search for each pattern is performed on the complete ASR translation. As seen on, the keyword tree implements individual nodes for each non-repetitive character forming a tree structure that optimizes the search of multiple words. The initial node is the root. Similar words follow the same initial node and path on the tree structure and create a new nodes and branches for any words that diverge from the established character nodes. The keyword tree is shown on Figure 3 and described by the search set P.

$$P = \{keyword, search, keywest\}$$

Notice how new words can be derived from old words optimizing the search space. The word Keyword contains the same root as Keywest, however only four additional nodes on the tree represent. Each time a search s performed all the words on the tree are considered in a single pass. The AC sting matching program will attempt to locate the patterns within the set $= \{P_1, \dots, P_k\}$ located in the input text string $S[1 \dots m]$ where $n = \sum_{i=1}^{k} |P_i|$ is the set of exact pattern matches. The algorithm has executes in two parts. First, the construction of the keyword tree; the second part is the search for the pattern in the input string S .

The construction of the tree begins with the root node by inserting an additional node for every character of the keyword. If the path selected ends before the end of the pattern, P_i is inserted. Additional nodes are inserted for the remaining characters of P_i. The letter i determines the nodes of the path and the end of each pattern. Each value of i is saved for each keyword inserted and saved as a terminal node. The numbers at the end of each keyword represent each ending node. As a reference, Table 1 denotes each variable used for the Aho-Corasick algorithm description. The search of a keyword pattern Pi starts at the root node following the path of characters as long as possible. Traversing the tree is controlled by three functions *goto, failure,* and *output.* The goto $g(q, a)$ compares each character of the input string S with the characters in the word tree starting from the root node and moving to the next node until the edge is found. If the edge is not found the function returns a zero. Otherwise, the goto function$g(q, a) = \emptyset$.

Table 1: Aho-Corasick Variable Description

Name	Variable
Pattern Set	$\mathcal{P}$
Test Document	S[1...m]
Set of Exact Pattern Matches	n
Text Document	S
Nodes of each path at the end of each pattern	i
Current State	q
Target character	a
Edge character	v
Tree Traversal Functions:	
Goto Function	g(q,a)
Failure Function	*f(q)*
Output Function	*Out*(q)

Therefore, the goto function $g(q, a)$ gives the state entered from the current state q by matching the target character "a". If the edge (q, v) is labeled by a, then $g(q, a) = v'$. The failure function $f(q)$ for $q \neq 0$, gives the state reached after a mismatch.

The output function *out (q)* tracks the set of patterns found when entering the state (Aho & Corasick, 1975). It is important to recall that in our particular experiment we match phonemes instead of American English Alphabet letters.

- ! & , . ? _ 1 2 aa ae ah ao
aw ax ay b ch d dh eh er
ey f g h ih iy jh k l m n ng
ow oy p r s sh t th uh uw v
w y z zh

Figure 4: American English SAPI Phoneme Set

Earlier we mention that the ASR converts its audio input onto phonemes that algorithm later uses to perform context searches. The Aho-Corasick algorithm suffered adaptation changes, but inherited the ability to perform phoneme pattern matching. The original genome DNA sequence patterns exhibit dissimilar characters in structure when compared to our phoneme representation. DNA representation consist of double stranded anti-parallel helix built by concatenating nucleotides consisting of Adenine A, Cytosine (C), Guanine (G), and Thymine (T). Note that a DNA pattern search requires a pattern word tree composed of patterns of single characters. Consequently, the Aho-Corasick algorithm was modified and adapted to support phoneme representations that require double character nodes within the word tree. The phoneme representation used is the Microsoft SAPI American English Phoneme Representation

("Microsoft Speech API (SAPI) 5.3," 2009). The string shown on Figure 4 represents the phoneme alphabet representation used by SAPI and our phoneme tree constructions.

In our particular case, we use SAPI phoneme conversions for both our input string $S\{1..m\}$ our patern matching set obtained by contextualizing our input string and further transforming its individual context to phoneme patterns that become $=\{P_1, \ldots, P_k\}$.

Phonetic Audio Indexing and Categorization and Search

The Queensland University of Technology and (QUT) participated in the 2006 NIST Spoken Term Detection. The task at hand was to locate English terms accurately in a given corpus of broadcast news and conversational telephone speech. The particular QUT system use phonetic decoding and Dynamic Match Lattice Spotting to locate the sought terms. The system consisted of two distinct stages, an indexing stage, and a search stage. The division of tasks allowed most of the processing to be performed offline line while the search will be done posterior to the indexing. During the indexing stage, phonetic decoding was used to generate lattices that would be inserted into a searchable DB. On the other hand, the search was done dynamically matching phonetic sequences with a target sequence using a Dynamic Match Lattice Spotting Technique. Phonemes where extracted using a Viterbi phone recognizer to generate the phonetic lattice. Tri-phone Hidden Markov Models (HMM) and a bi-gram phone language model were used during decoding while a 4-gram phone language was used for rescoring. The result was a collection of phones sequences that was stored in a hyper-sequence database.

During the search, the sought term is initially presented to the system and converted to its phonetic representation using a phonetic dictionary. If the word is not found a letter to, sound rules are used to estimate the corresponding phonetic and pronunciation.

As the target, phone pattern is decoded and used to find a match with the indexed pattern and in return, find matches that are identical or closely related. By using a Minimum Edit Distance (MED) phoneme recognition error is allowed by calculating the minimum cost of transforming an indexed pattern to a target pattern (Wallace et al., 2007).

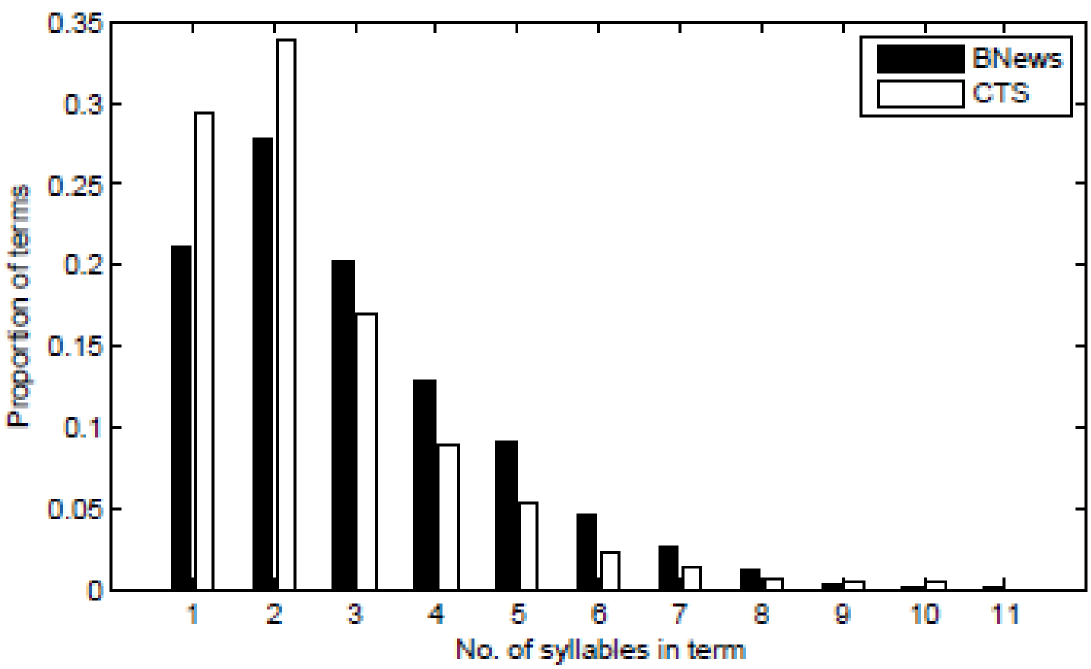

Figure 5: Histogram of Search Term Syllables Length
(Wallace et al., 2007)

The English evaluation consisted of about 3 hours of American English Broadcast News, 3 hours of Conversational Telephone speech and 2 hours of Conference Room Meetings. A total of 898 terms were given as search tokens for the Broadcast News and 411 for the Conversational

Telephone Speech Content. Each term consisted of a word with a varying number of syllables; however, most of the words contained 1 to 5 syllables a shown in Figure 5, but a few words contained 11 syllables.

The system was trained for speech recognition using DARPA TIMIT acoustic phonetic continuous speech corpus and CSR-11 corpus. About 120 hours of speech were used for the Broadcast News and about 160 for the Conversational Telephone Speech models. Letter to sound rules were generated using the CMUDICT 0.4.

The overall results showed that the best Phone Error Rate was 24% for the Broadcast news and 45% for the Conversational Telephone Speech. The ATWV was .22 for the Broadcast news and 0.8 for the Conversational Telephone Speech data. The Maximum TWV was 0.24 and 0.10 respectively for the two sets of data.

We can observe for this experiment that the QUT implementation of the phonetic search did not produce optimistic values when compared with the ATWV of the other participants. The authors commented that one of the difficulties of phonetic search is the large number of false alarms generated when searching short terms; specifically with terms that were 1 to 4 syllables long. Short phonetic terms can be hard to detect because they can become part of other words. When a phonetic term is small, its phonetic component becomes identical to phonemes contained in larger words. This generates false alarms, by retrieving sections of words that generate positive hits but correspond to a section of a longer term. With longer terms, the performance proved better, but the results are shattered by the large amount of short syllable words. The authors concluded that the system produced valuable spoken term detection performance, but performance improvement are necessary for verification and confidence scoring specifically on short terms if a requirement is to compete with LVCSR engines (Wallace et al., 2007).

Other companies have been experimenting with indexing audio. Some applications have been seen experimentally though YouTube and News Broadcast media such as Meet The Press. We collected video material from meet the press to gather a corpus to test the indexing and search of phonetic material

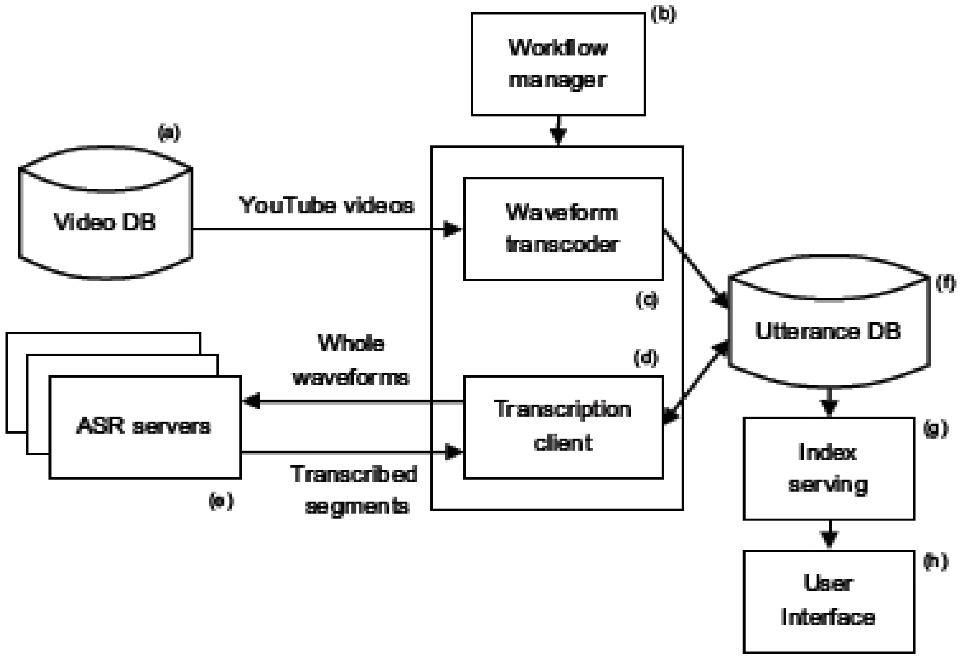

Figure 6: Google Labs Video Indexing Architecture (Alberti et al., 2009)

Recent work performed by Google on Audio Indexing Labs using Google Audio Indexing Technology (Gaudi), suggest the use of Spoken Term Detection (STD) to index videos using ASR Translation Information. Gaudi provides a richer search signal by providing the transcript of spoken content in the video. On research published regarding GAUDI, teams of researchers propose a system that indexes video time-aligned with words. Around the year 2008, the United States presidential election race had been using a video sharing service to promote their presidential candidacy for the election, creating a large repository of videos that they want the public to view. The demand was so large band YouTube created a separate to accommodate the election material. Most videos were an hour-long, rich in speech and sometimes presented crosstalk discussions between the candidates. Given the length of the videos, it was difficult for the user to search the information contained in the videos. The new information can be categorized by metadata, but such categorization does reveal the content of the video, perhaps just a clue of its content. Google aiming to simplify the task of the user, decided to create a tool that will index the audio of the videos semantic and the time-aligned with the visual content for posterior search. The user the user would interact with the interface that will allow him to navigate through the video material based on the content.

The developed system converts video by scanning periodically for changes through a video database, and if these occur, a *Waveform Transcoder* strips the audio by down sampling the content to 16 KHz, 16 bit linear signal, and stores in as separate utterances in a DB. The audio was stored with the least amount of compression since 10% degradation may occur on WER just from compression alone. The stored utterances serve as an input to multiples ASR engines that will create the transcript while discarding music and noise. The ASR converts the audio using a multi-pass strategy where only the best-scored utterances are stored. The system was trained using the 96 and 97 DARPA Hub4 acoustic model training sets and the Hub4 CSR language model training. The result is a transcription that is time aligned with confidence weights for each word. The information retrieved is then stored in an utterance database that further indexes the utterances for retrieval (Alberti et al., 2009). The research team recorded about 10 hours of material from candidate websites to evaluate the proposed system. The system using 1997 Broadcast news test audio, the system yield 17.7% WER. When using the baseline system on the election data set the WER was 40.1%, however the transcribed videos did not appear poorly transcribed. The OOV measurement was 1.42 with the baseline system and now results were given for the test system; however, the authors emphasized on the importance of certain tokens that would not affect the OOV result, but for the user are important such as "Obama" or "Putin".

The experiment went further by expanding the trained vocabulary to include the baseline system corpus into the presidential and adding lexical terms generated by pronunciation and analogy, which performed well in conversion of words such as "super delegate" but did poorly with names. The following examples were given:

Barak :

phonetically found as:

" b_ae _r _ae _k" and " b_aa _r _aa _k"

Putin :

phonetically found as: " p_ah_t_ih_n" and "p_uw _t_ ih_n"

These results are comparable with result obtained in our research tests and addressed using PDS to find data that is corrupted due to ASR errors generated by different environmental factors such as voice, speed, noise within others, generating different phones for a single sample word. The resultant adapted system obtained a 36.4% WER and an OOV rate of 0.5%.

The videos duration varied from 14 seconds to less than an hour. The system uses scalable Google infrastructure not disclosed. Figure 6 describes the architecture used on this particular research.

Phonetic Search

After evaluating the results of a phonetic search on both data sets, we can infer that most of the discrepancy found was related to the synthetic voice used to convert voice to phonemes automatically. Although we were using Windows 7 SAPI for speech recognition and voice synthesis, the phoneme conversions from each system were from time to time different.

Analysis of the data demonstrated that the word search was finding not only matches for the specific test word, but also words that contained the root of the sought word, a task that the phoneme counterpart omitted. Detail analysis of the ASR translated data, revealed that the phoneme set had the correct phonetic information to describe the word sought phonetically, but its phonetic translation within the search application had errors caused by the speech synthesis; the phonemes used to construct an utterance were phonetically accurate but syntactically erred, therefore inserting allophones.

Algorithms using Metaphone and PDS were created to fix the repeated phonetic errors die to the ASR converters and added to the search interface and indexing of the ASR conversions data to improve the results.

Contextual Summarization

As part of the indexation process, we add to the original ASR transcription contextual summarization that can be used to describe the content at a higher level such as metadata does describe objects in WEB searches. The process of converting dialog of multi-user content by running it trough an ASR is studied by many. It is known that the process generates a significant amount of error in the translation observed as WER. In this transformation and classified as errors, words are inserted that digress from the original content making the textual translated document dirty. Why not add positive content to the translation that describes in words that are semantically similar but different in syntax. With the help of summarization, we will be able to supplement the original content with ulterior meaning based on an electronic summary of the content. We call this process Context Based Indexing. As the translated text emerges, we process its content using contextualization tools found from Yahoo, Google, and Calais.

The Yahoo Term Extractor API, permits content analysis service that takes a block of text along with an optional helper phrase, and extracts relevant keywords based on the subject matter text provided as input. Yahoo Term Extraction allows users to integrate this ability into their own application and perform content analysis free. In return a significant list of words or phrases are extracted from a larger content submitted previously ("Yahoo! Developer Network - Developers Resources," 2009). The API expects a string with the text to be analyzed and an internet connection.

Open Calais web service automatically attaches rich semantic metadata to the content you submit. Using natural language processing, machine learning and other methods (OpenCalais, 2009), Calais categorizes and links your document with entities (people, places, organizations, etc.), facts (person "x" works for company "y"), and events (person "z" was appointed chairman of company "y" on date "x"). By using artificial intelligence (AI), natural language processing

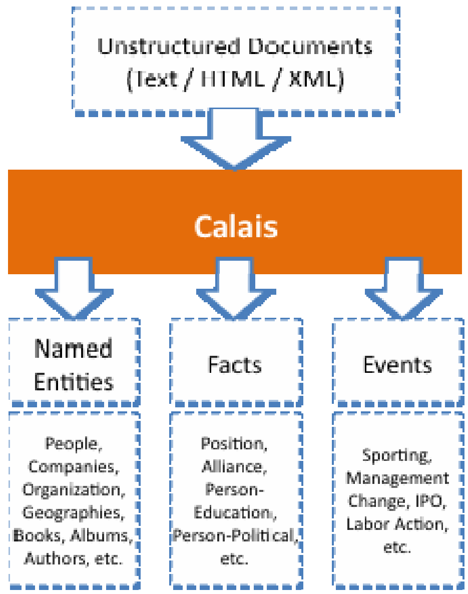

Figure 7: Open Calais

(NLP), machine learning within others, analyses the documents submitted and returns the facts and events hidden within the text. These tags are delivered to the user to be incorporated into any application. Figure 7 is borrowed from the Calais web site; it summarizes visually the contextualization goals of the API tool. Open Calais returns terms organized by Named entities, Fats and Events. Within each clasification Calais provides tags that are relevant to the original document and further sumarize its content. Perhaps, the contextual tems can be used to sumarize video content.

We use the information obtained from both Yahoo and Calais to provide summarization text that is stored and indexed into the database to provide instant con textual information about the dialog analyzed.

Figure 8: Yahoo Term Search Results

To highlight a small example of the capabilities of each API, we selected a small text that was translated using the current ASR and submitted to Yahoo and Calais without alteration. The following text on Figure 9 is an original ASR translation using windows 7 SAPI on a video recording of Bill Gates at commencement speech at Harvard University. We will use this extract to demonstrate the summarization results from a system query requesting context information.

Members of the Harvard corporation and the board of overseers members of the fact of the parents and especially the grassroots I'd been meaning more than 30 years to say it's down I always told you I'd come back and get my degree if a mundane honor of this honor of the changing my job next year and will be nice to finally have a college degree on my resume I applaud the graduates for taking a much more direct route to your degrees are not my part of this capital crimes involving Harvard's most successful dropout I guess that means the valedictorian of my own special class ID and the best of every one bail I also want to be recognized as the night not Steve Ballmer to drop out of business school if I'm a bad influence that's why I was invited to speak at your graduation invites opening your orientation if you are you might be here today Hubbard was a phenomenal experience for me academic life was fascinating I used to set an unlocked classes nine and even signed up for and unlike most rapid island up and ran for in our house are always a lot of people in my dorm room reading nine discussing things because everyone knew that I can worry about getting up in the morning at a liking to be the leader of the antisocial room weeklong each other's way of balloting of rejection of all those social people Ratcliffe has replaced the LAN and more women out there and most of the guys from outside clients economies and offered me the best clients if you know and I mean that's when I'm Linda sad lesson and improving your logs doesn't guarantee success

Figure 9: Bill Gates Commencement Speech ASR Conversion.

The text in bold as shown in Figure 9 is the part of original text that was correctly translated by the ASR. The green or lighter text is all the errors the ASR induced as part of the conversion. The resultant WER is 38.09% based on estimates using the CMU

After submitting the text programmatically to Yahoo Term Search, the reader can observe that the summarization tool does a good job of providing contextual information regarding the topic of the dialog. The results are shown in Figure 8 all not all correct since the text submitted already caries OOV words from the translation. Words such as “hubbard” are originally Harvard.

Similarly, Open Calais API provides the same service with different responses. Calais does a better job in categorizing the terms found by topics, Social Tags, Entities, and Events or Facts. The Figure 10 depicts the results from Open Calais. The reader will find that the categorization of terms found in the dialog is better organized and does provide in most cases, accurate results even when with OOV words are inserted due to ASR inaccurate translations It can be seen that information retrieved relates to education, United States and Harvard University that describe the small commencement speech. These words can be provided as descriptor terms that provide additional words to describe the context of the speech. Therefore, on a search to locate this video a small group of terms will describe the video as it becomes available for immediate search instead of the entire video/audio translation. Indeed, the example presented is short, perhaps the usefulness if this feature will become evident in large video/audio files stored in large amounts for search and retrieval.

The information obtained from these two summarization sites is used to create within the application contextual information about the dialog submitted, form a recent ASR dialog conversion. Then, we have terms that represent the context of a dialog, which are attached to the original dialog with the use of unique identifiers known as a Globally Unique Identifier (GUID). A globally unique identifier or GUID it is a special identifier used it in software applications to

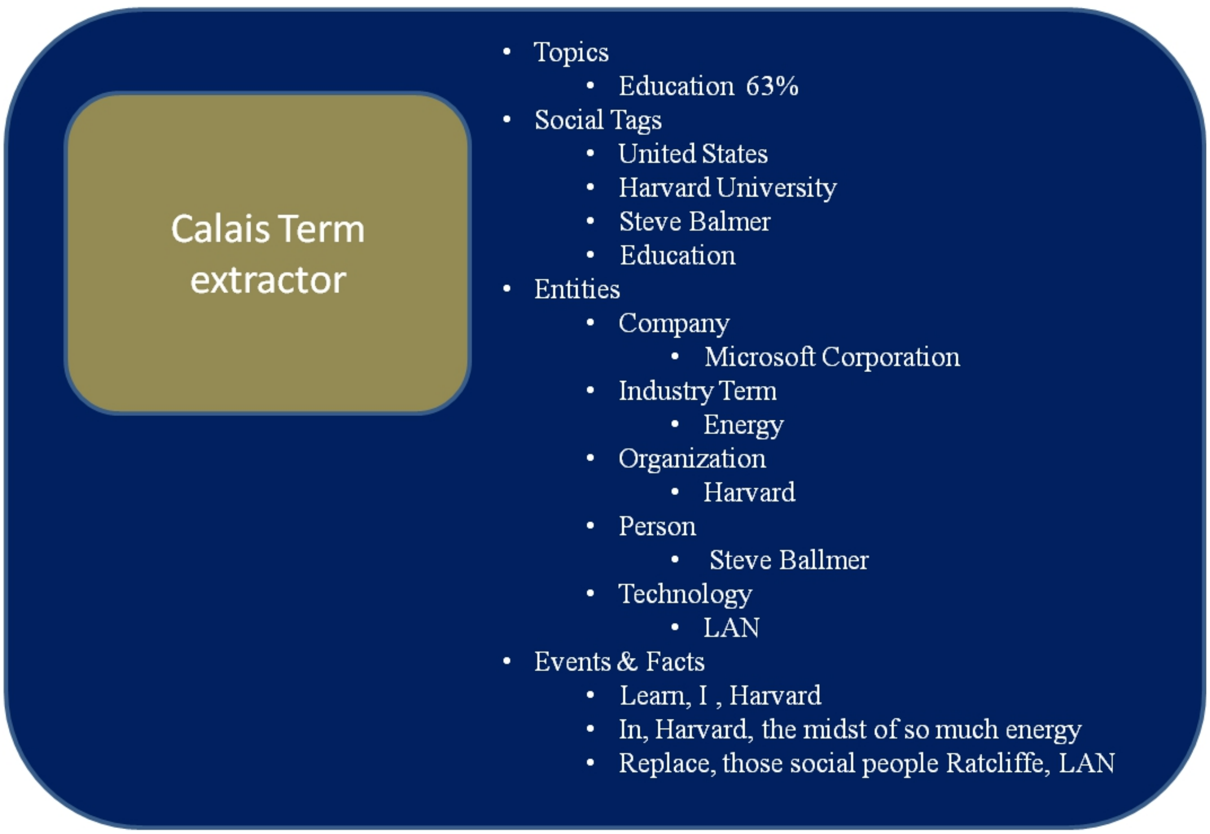

Figure 10: Open Calais Summarization Results

provide the reference number of its unique globally. The values represented by a hex of the civil string, 32 characters such as {21EC2020-3AEA-1069-A2DD-08002B30309D} and typically stored as a 128-bit integer. In our particular case, we use Microsoft's implementation of the

Universally Unique Identifier (UUID) standard for all indexing and unique key indentifying operations. The overall Contextual Summarization problem is summarized in the following chart (Figure 11). Notice the use of ASR and speech synthesis to obtain the phonemes for the summarization terms.

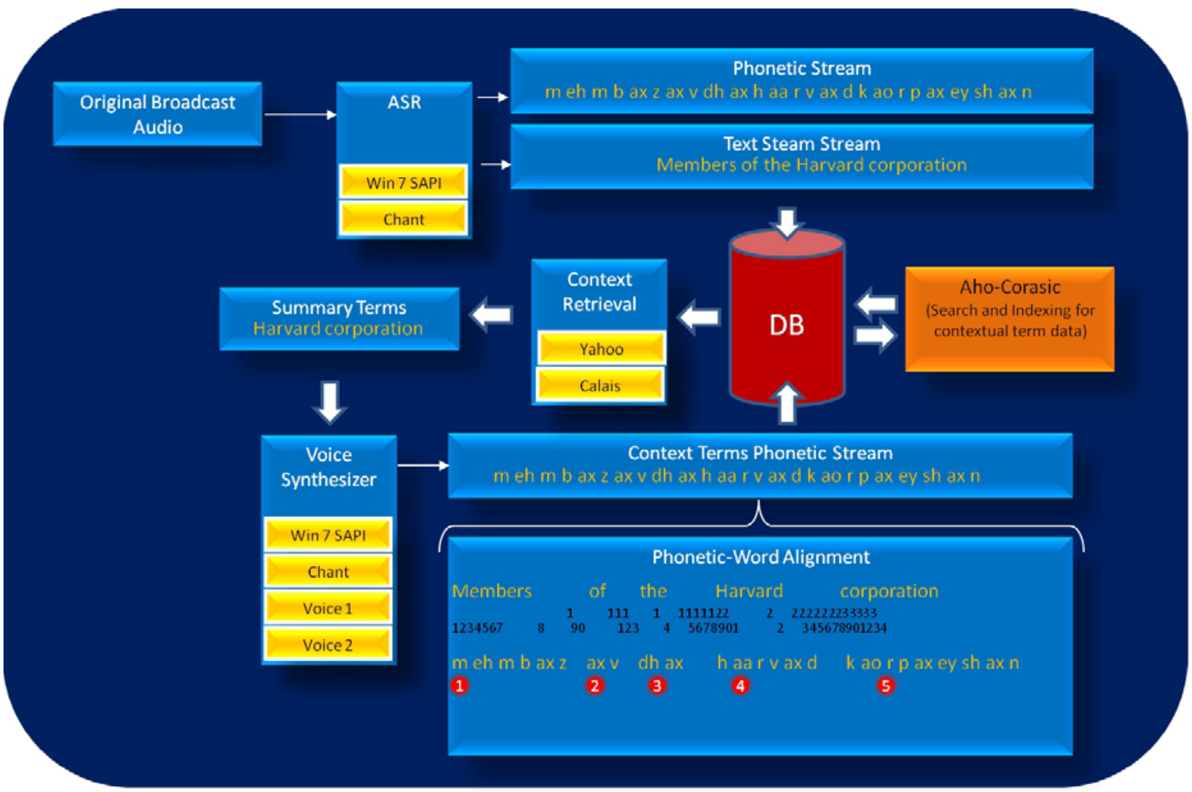

Figure 11: Contextual Summarization Process

The use of these summarizing API has been used in many applications. In a UCF, parallel project funded by NSF LIFELIKE the same approach was used to preserve memory of speech during an AVATAR human interaction. The entire interaction dialog was recorded and summarized to provide a later overview to the user. Furthermore, on future visits to the system, as the user is recognized, the Avatar surrogate will summarize the last visit and the information exchanged (Hung, Elvir, Gonzalez, & DeMara, 2009).

The following chapter better explains how the context based indexing and GUID identifier live within the database schema.

Thus, we address the tasks of storage and retrieval of conversational memory for a spoken dialog system; in this case, news broadcast video content.

More importantly, we describe two contributions: (1) a process for determining the prevalent contexts in transient and current conversations, and (2) a prototype system for accomplishing the aforementioned tasks. For the purposes of this discussion, we will focus on a broad, finite domain of dynamic contexts. Within this scope, we refer to a *conversational context* as the set of topics suggested by the utterances of all parties involved in the dialog. Moreover, we specify a *dynamic context* to be an abstract construct with a predefined structure, but whose possible range of attributes are not known a priori. The sections to follow will discuss the procedure used to populate this structure, as well as the role of the dynamic structure in maintaining conversational memory.

Through the memory interfaces at the topmost layer of the stack, the architecture services requests for recalling events that have been contextualized and stored in a database. Our implementation of memory interfaces is in the form of loosely coupled services. Weick (1976) first introduced loose coupling as a design pattern in which the knowledge of one class with respect to another on which it depends is limited to include only the interfaces through which they interact. In our case, the loosely coupled interfaces hide the implementation of processes internal to the memory architecture from audio/video indexing systems that might use it to store or retrieve content. At the same time, they allow communication to occur between the memory architecture and systems that use it.

CHAPTER THREE: CONTEXT BASED INDEXING

The contextual information is converted using speech synthesis. Most important, with this process we describe two contributions: (1) a process for determining the prevalent contexts in transient and current conversations, and (2) a prototype system for accomplishing the aforementioned tasks. For the purposes of this discussion, we will focus on a broad, finite domain of dynamic contexts. Within this scope, we refer to a *conversational context* as the set of topics suggested by the utterances of all parties involved in the dialog. Moreover, we specify a *dynamic context* to be an abstract construct with a predefined structure, but whose possible range of attributes are not known a priori. The sections to follow will discuss the procedure used to populate this structure, as well as the role of the dynamic structure in maintaining conversational integrity and storage.

The architecture services requests for recalling events that have been contextualized and stored in a database are addressed by a separate application. Our implementation of the dialog memory interfaces is in the form of loosely coupled services. Weick (1976) first introduced loose coupling as a design pattern in which the knowledge of one class with respect to another on which it depends is limited to include only the interfaces through which they interact. In our case, the loosely coupled interfaces hide the implementation of processes internal to the memory architecture from audio/video indexing systems that might use it to store or retrieve content. At the same time, they allow communication to occur between the memory architecture and systems that use it.

In a previous chapter, we mentioned briefly how the ASR translation is converted into text and phonemes. These utterance extractions from the original audio content are converted into text with the help of the Windows 7 SAPI compatible ASR. The resultant text is dirty; it is contaminated as seen on Figure 9, with OOV words as a result of the ASR translation. After the conversion takes place, we export the translated text and phonemes into a database, an utterance at a time where tag each utterance with the unique identifier GUID. Each utterance belongs to a particular dialog, which is composed of many utterances. As the ASR converts each utterance, we extract the phoneme and text information which is stored independently, but that is related to the original dialog by a dialog GUID and an utterance GUID.

Furthermore, the overall ASR translated dialog is submitted into a summarization service that in return provides content terms that summarize the original broadcast news dialog. Then, the summarization term information is stored also into the database with a GUID associated with it, but at the same time as related to the original dialog GUID. In brief, we have four different GUIDs to relate the different chunks of data, one for the original broadcast news dialog, another one for each utterance, and a final GUID that describes the dialog summary. With this GUID schema we provide reliable searchable information regarding the content of the original broadcast news dialog through phonemes, text, content summary and the location of each word went and the original content. Further information can be provided with simple distance calculations, such as phoneme location respective to word, word location in respect to the entire dialog as well as position and frequency information regarding any phone, word or utterance that are recorded into the system. We do not provide information regarding each speaker within the dialog because we cannot identify each speaker accurately with the existing technology.

This is extremely useful information since we are also want to track the back the position of each word with the original video regardless of the alignment of the words translated with the utterances in the video. The next figure helps to understand the relationships between the unique identifiers and the data.

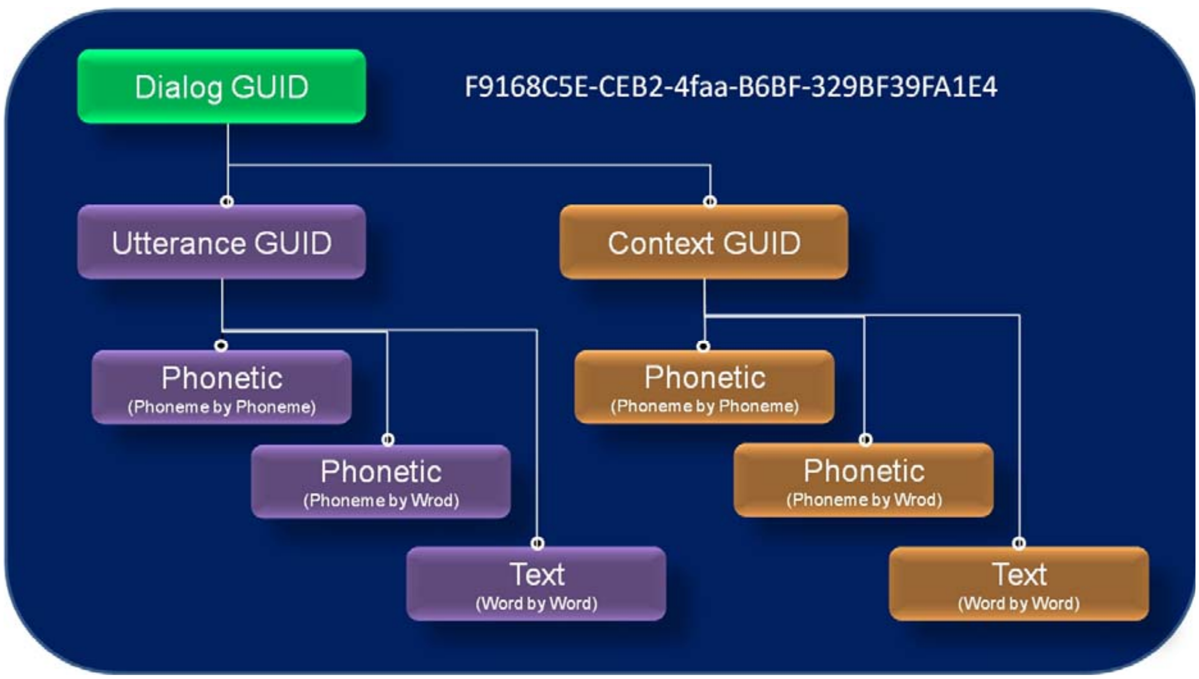

Figure 12: Database GUID Indexation and Assignment Schema

All information posted a spare is time stamped to further assist query/search process. Take the information regarding the search of phonetic data in relation to text data that is done with the help of the Aho-Corasick algorithm imported from genome pattern matching field.

Earlier we described the Aho-Corasick pattern-matching algorithm. Most of the pattern-matching algorithms use a moving window to find patterns in DNA sequences. Quite a few algorithms have grown into intelligent mutations that solve DNA sequence searching. Interestingly enough, phoneme representations accumulated in a string compare favorably to a DNA sequence when it comes to find possible methods to search for patterns. The phoneme strings that are extracted from an ASR translation can be imagined as a finite set of characters with a beginning and an end, but with no distinction between utterances. It is here where ASR translations meet genome pattern matching algorithms.

Pattern matching seeks the occurrence of a particular pattern or characters in a large string of text. Exact pattern matching searches all the occurrences of a pattern of ***m*** characters in a test of ***n*** characters based on a finite alphabet set $\sum$ of size σ

$$m\ (x=x_1, x_2, x_3 \ldots, x_m)$$

$$n\ (y=y_1, y_2, y_3 \ldots y_n)$$

I general, Pattern-matching algorithms use a window to scan trough the data in search for the pattern within the window known as an *attempt*. Alignment of the window is crucial therefore aligning the left side of the window with the text is first followed by matching the subsequent characters in the window.

As the match fails, the window traverses throughout the entire text in search of a pattern. At this point, the algorithm varies and it becomes specific to the particular research and pattern-matching application. Hundreds of pattern matching algorithms are available that perform better or worse based on the pattern length, periodicity and alphabet size. There is two phases to the pattern-matching, the preprocessing phase and the searching phrase (Thathoo et al., 2006). The preprocessing phase prepares the document and the window to minimize the search effort in phase two. Phase two attempts to find the pattern or patterns minimizing the time of the search and providing maximum efficiency.

Known algorithms that cater to improve the shift value are for example Boyer-Moore (Boyer & Moore, 1977), Quick Search (Sunday, 1990) and Berry-Ranvindran (Berry & Ravindran, 1999) within others. Specifically, we use the Aho-Corasick (Aho & Corasick, 1975) that uses a word tree instead of a traversing window. It can perform a search for multiple patterns at once based on a tree structure where the nodes represent the symbols of the patterns searched.

For the purpose of this research, we favor towards the use of the Aho-Corasic pattern-matching algorithm because its ability match multiple patterns in one data pass. Other genome algorithms above-mentioned, did not excel in this particular feature, however are excellent choices for DNS pattern matching research.

As we discover a need to recall, organize & index summarization data back into the original ASR phonetic translation, Aho-Corasic can search multiple phonemes extracted from the summarization in one pass. However the Aho-Corasic algorithm was modified for the phonetic alphabet as described in Figure 4, where the new alphabet size $\boldsymbol{\sigma} = 40$ and alphabet characters are in groups of one and two characters according to the phones syntax provided by SAPI under CHANT callback functions.

Chant software allows us to use SAPI functionality and the power of Speech Recognition and Speech Synthesis. It further extracts the phonemes from the utterances automatically as the speech is processed. It provides the ability to read audio files from different formats into the Speech Recognition or Synthesis. Most important of all, makes it relatively easy by providing a callback functionality from mayor ASR and Speech Synthesis manufactures.

By using the callback functionality, we are able to convert the contextual information retrieved using Open Calais and Yahoo term search into phonemes and use this information to search the broadcast news phonetic equivalent and determine if the phonetic information found could be matched; further determining if the summary term was a fair description for the particular video.

We were very satisfied with the performance of the gnome pattern-matching algorithm. Its ability to match occurrences of more than 200 phoneme groups from 5 hours of video in less than a second is remarkable. A similar SQL query onto a database would require multiple passes for each phonetic group and external programmatic filtering to provide statistical information about the search.

The subset of the context phonetic data that better matched the original ASR transcription can used to describe the original video content at a higher level, such as the metadata used to describe objects in WEB searches.

The use of the Aho-Corasic algorithm provides us with critical phoneme positional and frequency information that allows us to trace the summary terms back into the original phonetic stream. It is used to locate the words representations within the video; in spite of everything, the video is only an uninterrupted phonetic sequence of characters organized in a string and indexed by character. Therefore, positional information of the summarization terms can be easily traced by counting each character in a long array that holds the phonetic version of the original video aligned with the text conversion.

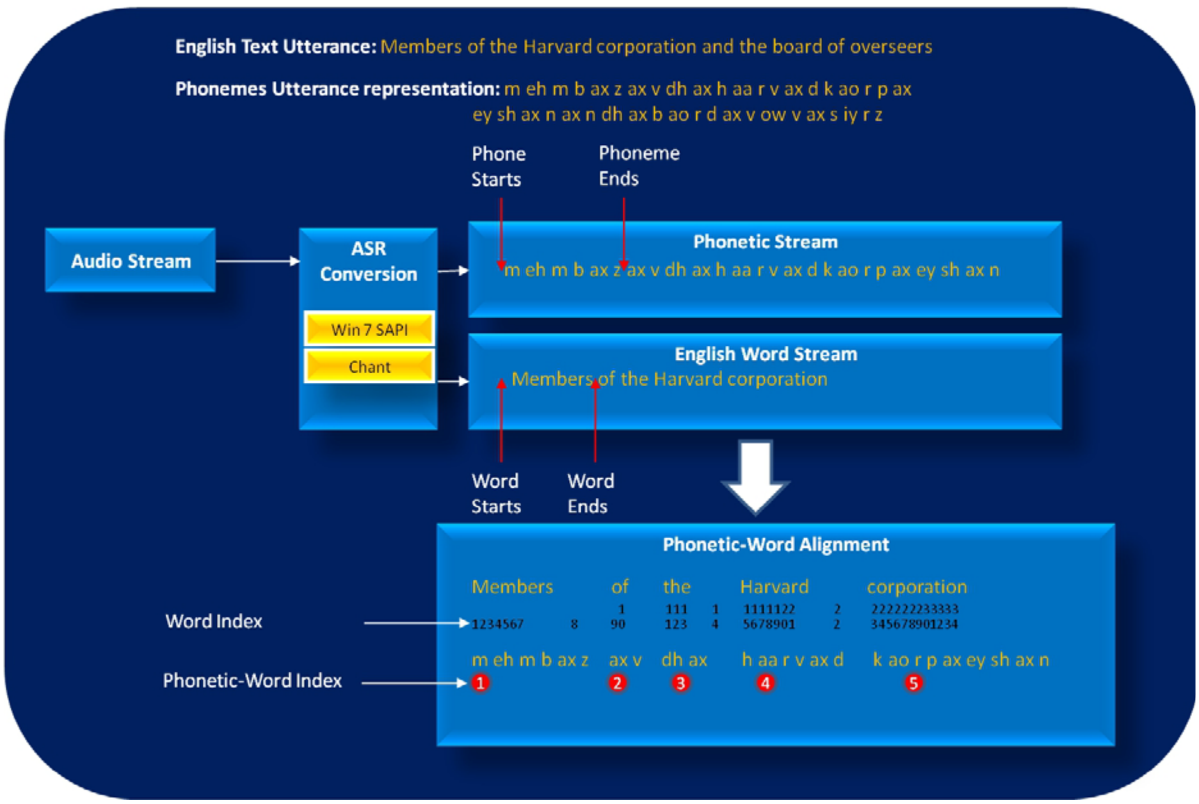

Figure 13: Phonetic-Word Alignment Algorithm

The alignment of the phonetic and word streams is intricate and an interesting problem we needed to solve. A group of researches envisioned a method of using ASR phonetic output in combination with "quasiphoneme" and audio spectral analysis (Torkkola, 1988). The suggested approach does not need any signal analysis; it is based on the position of the phoneme.

A priori, we know that the phonetic utterance and its uninterrupted sequence, when compared with the textual counterpart do not align well. The characters of each although similar in syntax, are semantically different presenting time alignment problem. Research shows that conversions using a string-to-pronunciation conversion algorithm we'll keep the alignment of the original English words with the phonetic conversion (Justin & Philip, 1996).

To solve the alignment problem, we implemented a voice-tagged algorithm that will keep track on a separate index the words with a phoneme sequences. This algorithm was built to work transparently well the phonetic extraction was executed by the ASR. As the ASR converted the original text into phonemes representing the original score, a parallel conversion to a word was also taking place. Every time the phone and was detected in parallel verification of the termination of a word was also being done, therefore words and phone groups could be aligned. By tracking the end of every word an index can be created to track the corresponding morpheme that represents the word. Then, each word has a phonetic representation that can be used to align both streams since each word itself is aligned serially in the audio by time and position.

Figure 13 better describes the phonetic-word alignment algorithm. It can be seen that two separate indexes are kept. The word index us used to locate each letter as referenced on the original text. The Phonetic-word index is used to keep track of the phoneme groups that compose each word. All information regarding the conversion of text is stored following the original GUID schema.

Figure 14 shows the interface that converts the audio content to ASR, contains all indexing and data conversion needed for the system operation. The Phoneme extraction and DB Indexing Utility reads a video/audio file, and converts it to the phonetic and word content and indexes it to fit the proposed DB schema.

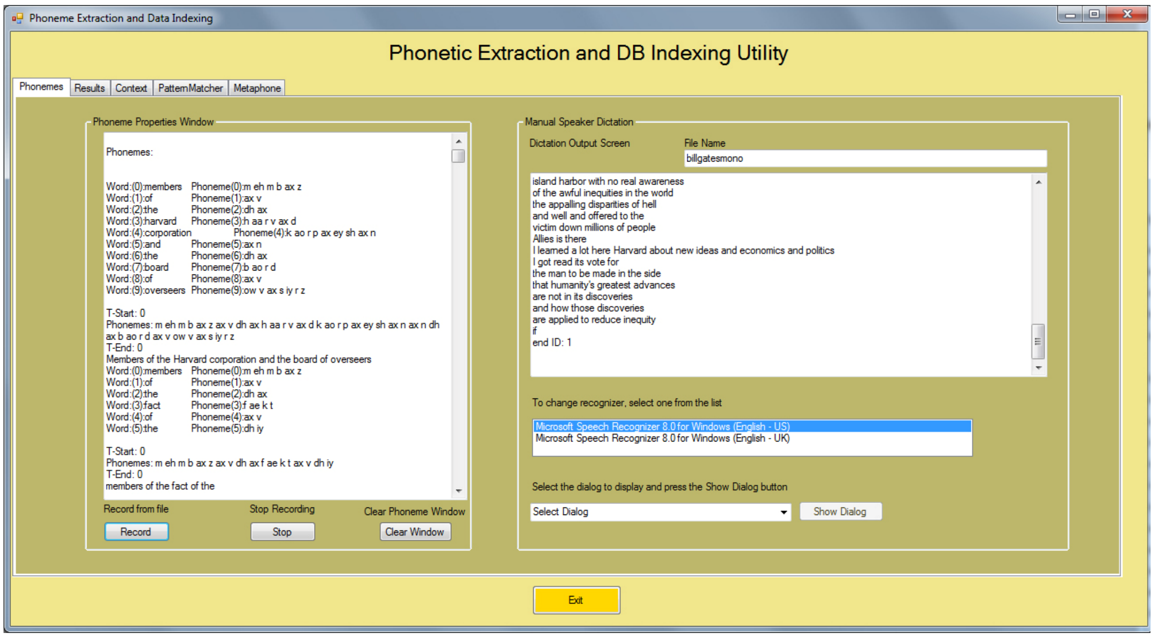

Figure 14: Phonetic Extraction and DB indexing Utility

This interface also performs the extraction of the video context, and also converts the context to phonemes and stores it in DB. Moreover, the same interface contains the utilities that provide the Metaphone indexation of the corpus and error rate calculation. The interface provides functionality necessary prior to any search operations. In a live system these operation will be scheduled round robin as new videos are submitted into the system for indexing, therefore categorizing all new data, and making it available for search automatically.

CHAPTER FOUR: PHONEME DISPARITY SEARCH

In this section, we analyze the Phoneme Disparity Search (PDS) versus a standard word search. This algorithm is the evolution of a phonetic search and observation made on 5 hours of video and phoneme test queries using over 100 words. The initial research hypothesis was to prove that a phonetic search could provide increased word spotting when compared to word search, because morphemes represent the sounds of words as opposed to words that are an English language representation of the sounds.

Word spotting is a reliable detection of a word in a specific speech utterance in this case converted into phonemes using a standard Windows 7 SAPI ASR. The goal of the PDS phonetic search is perform word spotting on our own video data corpus by exploration of different methods using phoneme search. The most commonly method or word spotting is using HMM, however it has its own troubles as the collection of non-keyword speech; perhaps more important is that HMM do not directly maximize the keyword detection rate (Wollmer et al., 2009). No evidence emerged that HMM was used for phonetic spotting.

However, with PDS we take a different approach; specifically, we try to minimize the errors caused by ASR speech synthesis of a word and implement the deficiency into a model that will overcome such insufficiency and augment a normal phonetic search with the capabilities of a wildcard search for unstable search conditions where the appearance of dirty phones is ignored.

We establish a bases line using standard words searched by obtaining a list of word that are known prior to the execution of the algorithm. The set of words or bag of words, selected for this particular system present difficulties inherent from imperfect ASR conversions as well as differences in length suggesting Class A, B and C word lengths that also affect the performance of the proposed algorithms. The suggested bag of words characterizes the 98000-word corpus compiled from video ASR translation. The alignment mapping of the video is done by referencing each phonetic character position back to the original word and its position in the video original video stream as explained in Chapter 3.

Our tests demonstrate that in situations where the ASR translation is corrupted due to intrinsic conversion errors, phonetic searches can provide additional insight and improved retrieval results. PDS uses different combinations of phonemes to augment a single search based on the assumption that a mix of speech recognizers will corrupt the search. By using different speech vendors, each query is then synthesized by different voices, each generating their own conversion. When each phoneme conversion is compared, the most probable errors are replaced by wild cards within the search improving the phoneme search dramatically. Further enhancing the capability of PDS, the phonetic structure is analyzed and cleaned of repeated phones caused by random ASR conversion using Windows 7 ASR.

PDS attempts to accomplish a "sounds like" search, rather than using and English Language interpretation of the sounds. We know a priory that all ASR conversions are obtained from sound files that represent each video. The sound files are processed many times to provide multiple ASR translated versions of the words into Microsoft SAPI phoneme format (Figure 4), to increase the possibility of a word altered by ASR conversion. In fact, we do video transformation duplication because the errors caused by ASR and speech synthesis translation which generate different phoneme sets for a single word. Therefore, a single word represented by a morpheme is composed in part by a different phone that causes confusion.

To identical words in American English language my look the same, but their phonetic interpretations can be different due to a single phone such as “ax”. This phone is highly confused by the Windows SAPI translations of corpus material. Nevertheless, other errors such as unnecessary repetitions of phones further corrupt the corpus material.

On Table 2 we show the word “president” as an example of its conversion as it is to be found phonetically in the corpus. The reader can notice the similarity of the different phonetic variations for a single Morpheme.

Table 2: Different Phonetic Interpretations Using Windows 7 SAPI ASR

Word English	Correct Phonetic Translation	Morphemes as found in ASR Conversions
President	p r eh z ax d ax n t	p r eh z z ax d ax n s iy
President	p r eh z ax d ax n t	p r eh z ax d d ax n t t t
President	p r eh z ax d ax n t	p r eh z ax d d ax n s iy
President	p r eh z ax d ax n t	p r eh z z ax d ax n t
President	p r eh z ax d ax n t	p r eh z ax d d ax n t s
President	p r eh z ax d ax n t	p r eh z ax d ax n t t t
President	p r eh z ax d ax n t	p r eh z ax d ax n s iy
President	p r eh z ax d ax n t	p r eh z z ax d ax n s iy
President	p r eh z ax d ax n t	p r eh z ax d d ax n t t s
President	p r eh z ax d ax n t	p r eh z ax ax d ax n t

However, in a phone-by-phone comparison, each morpheme is unique but its semantic meaning is the same: “president”. Further inspection can reveal that only one morpheme is correct, all the morphemes in the column to the right have an error. A typical error found is the repetition of phones at the beginning and ending portions of the morpheme, such as “p r eh z ax d d ax n t t t”, where the phone “t” is repeated 3 times. An extreme example is the word “economy” that on a few speech synthesis conversions has produced “ih ih ih ih ih ih ih ih k aa aa aa aa aa aa n ax ax ax ax m iy iy”, however not of significant recurence. Such errors described on

Table 2, are widespread, it affects the successful retrieval of words since the phone error insertion happens randomly. Preemptive filtering could be applied during indexation using an electronic phonetic dictionary that runs a cleaning agent periodically cleaning the corpus, however we chose not to alter the ASR translated corpus and use it as a baseline. However, recent research in China demonstrates an increase in normalized sentences by processing ASR translation content. On this research, they selected 8000 sentences from an ASR translation and tried to find them in a baseline non-normalized set, and in a separate normalized set. The group finds that after “Sentence Normalization Effect” the results increase from 8.4% to 41.1% by pre-processing the corpus (Huang, Feng, Wang, & Zhang, 2010).

Other representative errors found are the repetition of the “ax”, “ih”, and “aa” phones; perhaps caused by mispronunciation, bad diction and environmental noise, as well as unpredictable ASR conversions it time. The task at hand gets intricate as we perform two separate conversions of original content to create the corpus. The first conversion is the ASR translation that leads to the creation of the main video corpus; the second phonetic translation is the voice synthesis, which converts to phonemes the context terms and the phonetic search terms. Each conversion, ASR and Speech Synthesis is capable of generating different errors as demonstrated in the example.

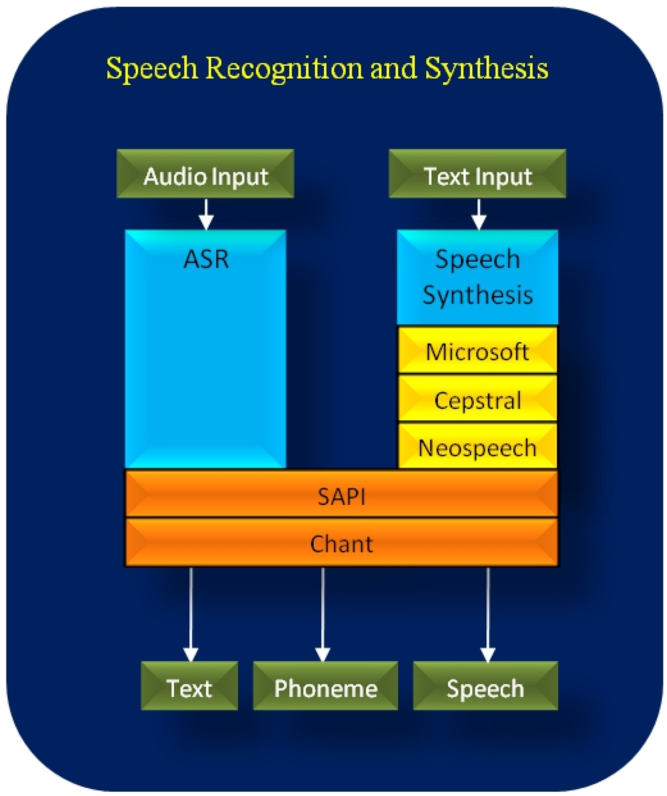

Figure 15: Microsoft SAPI block Diagram

Similar research done in other languages encounters similar problems where the phonetic pronunciation of certain words appear in the corpus dirty. For example the word “yes” in Chinese phonetic translation is found “as s ih d e” and “sh ih d e”, both incorrect translations. The group proposes the use of ASR phonetic data and Statistical Machine Translation (SMT) combined to reduce induced ASR errors. By using pronunciation as well as syntax checks they improve the reluctance of the phonetic search (Liang, Yonggang, Wei, & Yuqing, 2006) .

Interestingly, the errors vary on each side of the SAPI equation. The ASR and Speech Synthesis sides behave surprisingly different as we consider the phonetic translations. Although both sides are Microsoft branded, we wonder why they behave differently. We have spotted different phonetic translations that depend on the voice used for conversion. While searching through data, we found the use of different phones in a single word when converted using audio and ASR, than the phones that were created using a synthetic voice. Further, reproducing the same test by using a commercial synthetic voice from Cepstral ("Cepstral," 2010), the results matched the ASR phonetic example on test data. Then, we can conclude that in a real word scenario where multiple users will be posting audio, the standardization between ASR conversion and voice synthesis need to be observed, so that both sides of the translation have identical results on test data to diminish OOV word error rates. For our particular case, we know that the phonetic search had to address all these random factors.

A functional block for Microsoft SAPI is shown as a reference to clarify SAPI operation. The ASR component transforms Audio to text or phonemes and its counterpart the Speech Synthesis converts the opposite, text input to Speech using a synthetic voice.

To evaluate the changes we decided to add another commercial voice into the mix to observe the results of the two text-to-speech voices processing the test data. We used the standard windows 7 “Anna” voice and Cepstral “Allison” voice for out tests. We ran tests using both voices in search of a pattern that could provide insight for a solution. Immediately, we noticed that on each result, a specific phone was replaced by another depending on which voice was used for the conversion. The most common case is the use of the phone ‘ax” which often is replaced by “ah”, “er”, or “ih”. These results marked the beginning of a Phonetic Disparity Search Solution. We would incorporate as part of the search, the ability to replace these phonemes and find all the words matched within the context of resultant set of phones, a set not affected by the translation errors.

Table 3 shows the disparity from one speech synthesizer to the other. The differences are palpable, specifically with the phones “ax”, “ah”, “ih”, “ey”, and “er”. Notice the substitution of “ax” for “ah’ and “er”, a common practice when Microsoft SAPI is used single-handedly. The addition of other vendor Speech Synthesizers correct errors partially as it inserts different errors.

We then need also to overcome the errors inserted during speech synthesis; randomly the ASR tends to add repetitive phone sequences internally and maybe caused by the interleaving of all the threads running in the application. We recognize that the phone “ax” is used by Microsoft for many phonemes inserting errors during ASR translation. In a perfect ASR, theses phonemes will not be different if used to represent a word; however, the truth of the matter is that these substitutions are within the corpus and PDS considers these errors within the search design for Word Spotting correctly regardless of the voice or ASR used for translation.

Word Spotting (WS) search can be done using in vocabulary word (IV), or new to the system out of vocabulary words (OOV). Previous studies have shown improvements on both types of searches using a phonetic approach, however the WS process presents a higher error rate compared to ASR based WS in the context of IV word search. The research further explain that there is a set of phonemes that are more likely to get confused such as phones “b”, “bd”, “dd” and “gd”. Each confused phone is part of one of seven Metaphone groups (Arnon, Alon, & Savitha, 2001).

Similarly, we have our own confusion matrix of phonemes that are used randomly by the ASR and Speech Synthesis SAPI translation systems. However, the phones found are different from abovementioned research and shown below. We believe that the use of different vendors on ASR and Speech Synthesis generate different syntax for the American Phonetic Alphabet. The implications of the different phonetic alphabets for PDS are that the implementation of such algorithm will vary based on the vendor used to support ASR and Speech Synthesis. Nevertheless, the system shall support Microsoft based applications. Further tests can be conducted with other vendors in comparative test, however to considered due to cost.

Table 3: Phone Replacement Errors Caused by ASR and Voice Synthesis

Word English	Morpheme ASR Conversions	Morphemes Voice 1	Morphemes Voice 2
Economy	ih k aa n ax m iy	ih k aa n ax m iy	ih k aa n ah m iy
Economists	ih k aa n ax m ih s t	ih k aa n ax m ih s t	ih k aa n ah m ih s t
Economic	iy k ax n aa m ih k	iy k ax n aa m ih k	eh k ah n aa m ih k
Jobs	jh aa b z	jh aa b z	jh aa b z
Harvard	harvard	h aa r v ax r d	h aa r v er d
Inflation	ih n f l ey sh ax n	ih n f l ey sh ax n	ih n f l ey sh ah n
President	p r eh z ax d ax n t	p r eh z ax d ax n t	p r eh z ih d ah n t

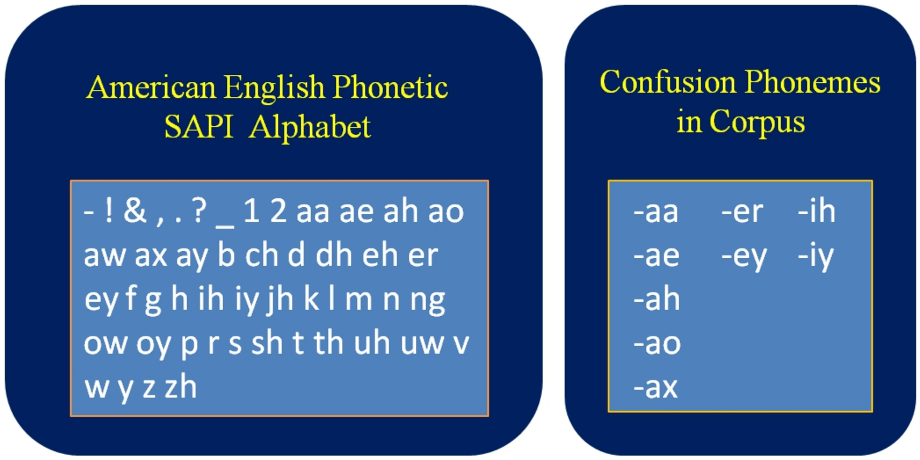

Figure 16: Confusion Phones fond in Corpus

The ASR translation and categorization provides us with two sets of data for experimentation, the ASR translation, and the conversational contextual words extracted from the ASR translation. Thus, the ASR translation is automatically converted to American English words and phonemes and stored separately. The contextual word information is also stored in word and phoneme versions separately with the aid of the speech synthesis. All stored material inherits OOV words from inaccurate conversion due to the imperfect source material and speech recognizers. We hypothesize that because the phonemes preserve the original sound of the word we can use phonetic information to expand the search further and hit related content where a word query would have failed due to a semantic loss at conversion. However, phonemes also carry conversion errors in translation, or Confusion Phones. These confusion phones are a set of phonemes that change randomly due to source changes in speech, causing confusion while searching because the multiple phonetic versions of a word. Therefore, an identical word can have different morpheme interpretations.

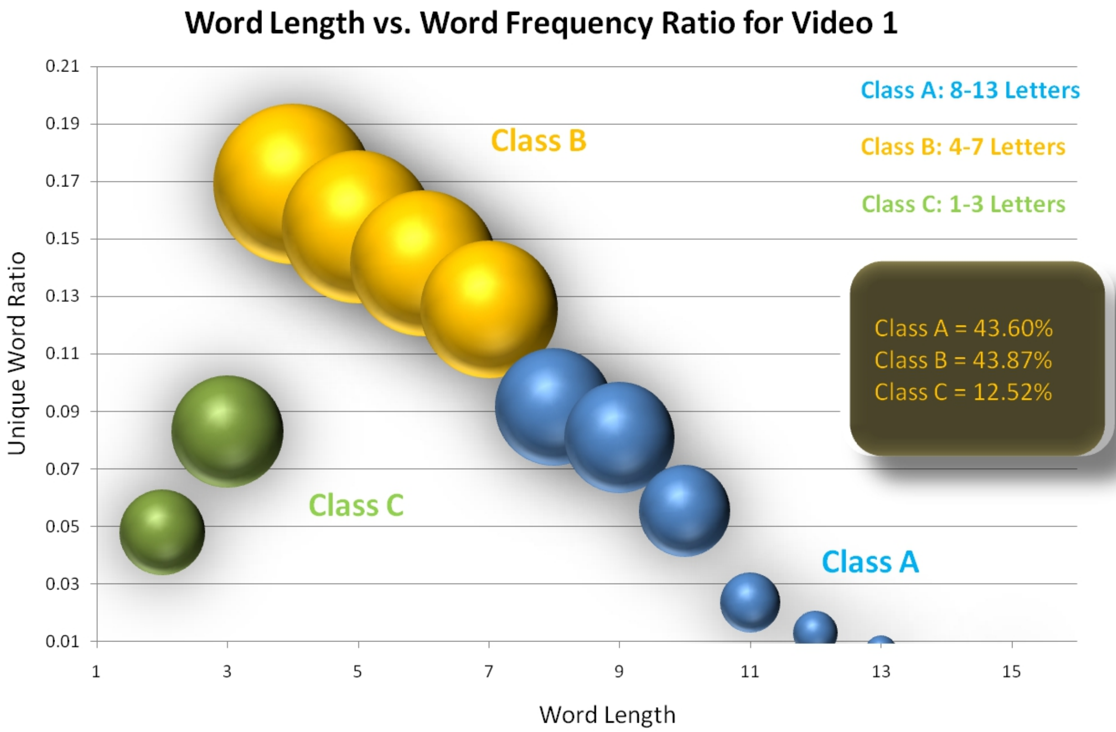

Figure 17: Search Term Length.

Initially, we tested the capacity of a distinct phoneme search versus a distinct word search using the smaller sized contextual word data. We selected a set of words that contained 4 to 8 letters and tested their correct phonetic translation repetitively to establish a baseline. As seen on Figure 17, most of the words in from the selected audio collected from video 1 is within Class A and B characters that include words such as “president” and “unemployment” that contain 9 and 12 letters and account for about 6% and 15% of the video 1 corpus accordingly. As we tested using straight queries into the database and searching for the words and the phonetic patterns, we discovered that the resultant phoneme search hits were less than the word search hits, where a phonetic search will succeed 18.3 % of the time and the word search 81.7 % of the time.

These calculations include de search for IV and OOV words within the 100-word test. The OOV words sub-set returned no results using a phonetic search. However, as we looked at the set of words found by the search that were highly frequent compared to the phoneme counterpart; we found a disparity within the word and equivalent morpheme. The test corpus contained words that while duplicate in the word ASR translation were different in the phonetic translation as shown at the beginning of the chapter. Further, words with four letters of less, when searched using phonemes tend to retrieve words that are longer than four characters but that contain the phonemes sought as a subset. Words that have four letters account for less than 8% of the vast majority of the content.

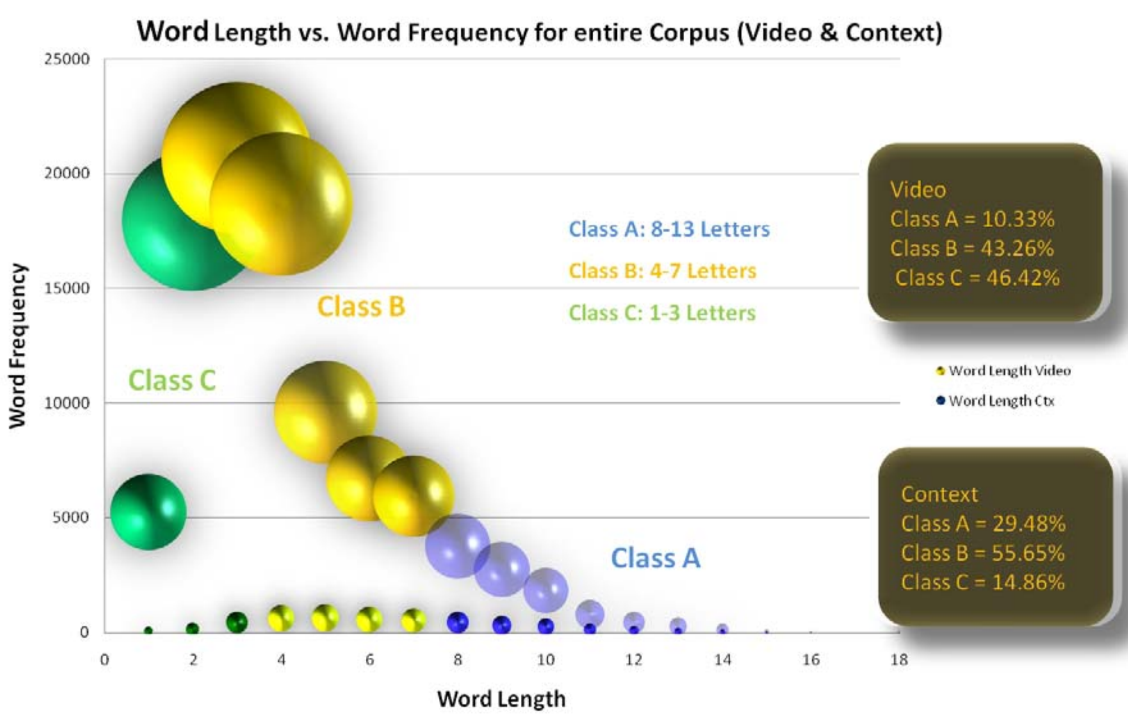

Figure 18: Search Term Length for Video and Context Dialog.

On a larger scale, tests performed on the entire corpus generated positive results produced similar results. As seen on Figure 18, Class C words that are less than three characters account for 46% of the video search space however not critical since words in this class are mostly consonants or pronouns within others and not generally used to describe a topic. Class A and B account for 53% of the search space, a region where PDS proves reliable.

On our corpus test results, we notice that, as the set of letters that compose a word gets smaller as the number of false positives increases. For this reason, we divided the words in three classes. Class A are the words that have between 8-13 letters a group that work well with most relevant searches. Class B is for words with 4 to 7 letters, and Class C for words with 1 to 3 letters. Class C is reserved for the group with the most false positive results since three letters start to appear as parts of larger words; exact matching works better for this group.

After evaluating the results of a phonetic search on both data sets, we can infer that most of the discrepancy found was related to the synthetic voice used to convert voice to phonemes automatically. Although we were using Windows 7 SAPI for speech recognition and voice synthesis, the phoneme conversions from each system were from time to time different.

Further analysis of the data demonstrated that the phonetic search within the words found, was matching words that contained the root of the word sought, a task that the word counterpart omitted. Detail analysis of the ASR translated data, revealed that the phoneme set had the correct phonetic information to describe the word sought phonetically, but its phonetic translation within the search application had errors caused by the speech synthesis; the phonemes used to construct an utterance were phonetically accurate but syntactically erred, therefore inserting allophones.

This result supports our hypothesis and our belief that there is not a one-to-one correspondence between a single phoneme expression and an ASR conversion of a word. We thrive in producing hits that the word search cannot perform due its relation to syntax rather than semantics. The phonetic search if modified to adjust for the errors generated by the ASR translation could be used to find not only the word sought, but also brothers and sisters of the same word, therefore retrieving related words that are also of topical interest and omitted by the word search.

Perplexed by the discrepancy, we experimented with different synthetic voices. Surprisingly, we find that other vendor voices from time to time convert a word to phoneme differently; an event that suggests which phoneme is part of an incorrect phonetic translation or allophone. To compensate for allophones programmatically, we initially tried phonetic conversions using different languages within a single Commercial-Off-The-Self (COT) speech synthesis voice. The resultant foreign language phoneme conversion, replaced some out of language phonemes by empty spaces. As we tried different American English voice synthesizers, we noticed that on certain test words, the phoneme differences were evident in their phonetic conversions. We found that there was an inherent phonetic disparity related to the different voices, but that such nuance affected the same phoneme or phonemes. Furthermore, in most test cases all synthesizers used converted words to phonemes identically.

Figure 16, is the outcome of the use of multiple voices. We were able to discover within our system a set of confusing phones to avoid in a search. All we had to do was to create a single pass algorithm that would avoid searching for the phones in the confusion matrix, but that would include as part of the search the healthy side of the conversion. If successful, we will have a Phonetic Disparity Search that would avoid the nuances of searching by phonemes and demonstrate that a phonetic can perform better than or baseline word search.

To test our approach we built a modifiable interface that permits modification on how the phoneme is used to perform the search. After a search word is inserted, the standard SAPI controlled speech synthesis engine would convert the sought word to the equivalent phoneme and use it to perform a search, however with the help of slide bars positioned on the User Interface the length of the phoneme could be adjusted as multiple searches were performed. The result of each search would be stored in a Database for posterior analysis. Furthermore, as each search was performed the voices used to perform the search could be exchanged, information that was all kept for analysis. The data obtained using the Context retrieval and indexing interface would be used to load the necessary audio and provide phonetic and word translations of the corpus for analysis.

Three hours of video material was recorded and processed multiple times to create a vast library of content. Duplicates of the videos were part of the content to add additional insertions of confusing phonemes into the corpus. A set of test words was identified, and a test case, each word would be searched four times using a standard phonetic search approach, a standard word query approach, and a PDS search. Each time a search was performed for a test word the PDS would be adjusted to two characters at a time by moving the slide bar therefore adjusting the size of the phoneme sought, i.e. p p r eh z ax d ax n t t t could be adjusted to eliminate two “t” at the end of the morpheme and pass the new morpheme as the new search.

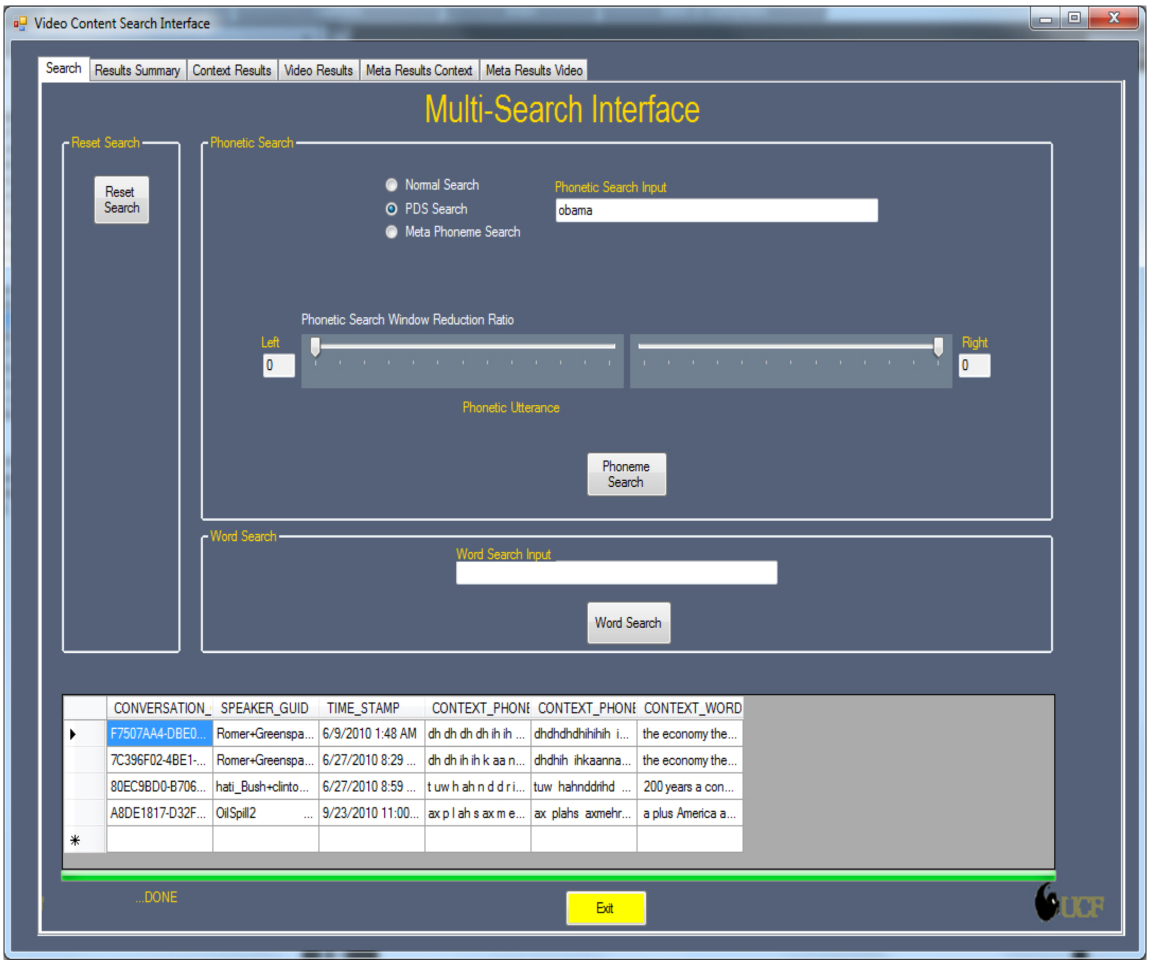

Figure 19: Multi-Search Interface

On Figure 19 we can inspect the Multi-Search test interface used to collect test data for PDS development. The program allows four searches: a baseline word search, a baseline phonetic search, a PDS search, and a Metaphone search. The former, is the topic of the following chapter and mentioned here for reference. This latest version of the interface does not have the voice selection since this was lost from older versions as PDS algorithm absorbed the functionality.

The first round of tests generated enough information to understand the evolution of the PDS algorithm. For the purpose of the process explanation, we will concentrate in the use of two words as examples, i.e. "president" and "economy". The Table 4: PDS Initial Results, it is a comparison between the result of a word search and multiple PDS searches using the adjustable bars to delete dirty phones.

	Phoneme	PDS Ctx	PDS Video	N-Videos	Total Freq.	Left PDS	Right PDS	Word
Phonetic Search	ih k aa n ax m	10	53	5	63	0	1	economy
	ih k aa n ax	14	59	6	73	0	2	economy
	ih k aa n a	14	59	6	73	0	4	economy
	ih k aa n	22	59	7	81	0	5	economy
	ih k aa n	22	59	7	81	0	6	economy
	ih k aa	23	64	7	87	0	8	economy
	ih k	161	454	9	615	0	10	economy
	h k aa n ax m	10	54	5	64	1	0	economy
	k aa n ax m	10	70	5	80	3	0	economy
	aa n ax m	11	71	5	82	4	0	economy
	aa n ax m	11	71	5	82	5	0	economy
	a n ax m	11	71	5	82	6	0	economy
	n ax m	26	99	8	125	8	0	economy
	ax m	129	795	9	924	9	0	economy
	k aa n ax	14	78	6	92	2	2	economy
	k aa n ax	14	78	6	92	3	3	economy
	aa n a	17	105	7	122	4	4	economy
	~~k aa n ax~~	~~14~~	~~78~~	~~6~~	~~92~~	~~2~~	~~2~~	~~economy~~
Word Search		17	72	7	89	NA	NA	economy

Table 4: PDS Initial Results

The initial approach was to delete all the extra characters from the phonetic search i.e. “ih k aa n ax m” where “ih” is missing the last phoneme in all searches. The reader should know, that the word search returned a total 89 word hits (Total Frequency Column) and established the new baseline for total frequency. Thus, a PDS search without any adjustment matched 63 records. Further adjusting the right side of the test morpheme by two characters, i.e. counting backwards from the end of the morpheme provided a slight increase to 73 hits on Context and Video translations combined. The best result was obtained using a Right-PDS value of eight that returned 87 hits, 2.2 % less. However, as the Right PDS value is adjusted beyond eight the value Total Frequency value spikes to 615 hits. At this point PDS is ineffective, the few phones available for the search have become a small subset of the morpheme that can be found in 615 words where 87 words are relevant, and 528 are not.

As the search is modified from the left, similar results are obtained for the baseline, 64 hits with no adjustment and a best of 82 hits with a left PDS value of six. Higher PDS values spike the total frequency value to 128 and 924 with values of left PDS of eight and nine respectively.

From the initial test case, it can be concluded that not all the phones are needed for an accurate search; a subset of phones can generate reliable matches. Values of Left PDS and Right PDS in general are stable below values of three for phonetic words of length 8 or more.

We can now formulate a PDS algorithm as the following: Initially transform the word sought to its phonetic equivalent using the speech synthesizer. Extract from the morpheme the phonemes from the confusion matrix, i.e. “ax”, “ih”, and “aa” if found in the morpheme. Then, create a search morpheme that replaces with spaces the deleted confusion matrix phones. Delete the last two characters of the new morpheme. Query your test database by using the updated morpheme, by replacing the spaces by wild card search tokens. Process the query. The query for the word "economy" after the morpheme transformation should be "%__ k __ n __ m%" instead of "%ih k aa n ax m iy%". Test data can confirm that PDS search when compared with the baseline, the percent change is 70.9% better than the word search for the word "economy" and 250% better with other test data to be shown in the Results and Analysis chapter. However, PDS does not always work; it is then when different methods need to be used concurrently to evaluate that method that yields the best Word Frequency. The following Double Metaphone Strategy complements PDS and is the topic of the next chapter.

PDS Algorithm Definition

The PDS algorithm emphasizes on the substitution of Confusion Phonemes by a wildcard character as it cleans the phonetic stream for repetitive phonemes not common except for "aa". The output pattern is the outcome of the algorithm; it includes the repetitive wildcard pattern embedded within the surviving characters from the wildcard character substitution. The output pattern is to substitute a Database SQL Query that shall search for the pattern in the corpus.

We define the Search Array, W, containing a phonetic array of characters as:

$$W(i) = \{\rho_1, \rho_2, \dots, \rho_\eta\}\ ;\ 1 \leq\ i \leq \eta, where\ \eta\ morpheme\ length \tag{7}$$

We define the Output Array, O, containing phonetic characters and wildcard characters as:

$$O(j) = \{\rho_1, \rho_2, \dots, \rho_j\}; 1 \leq j \leq \mu, where\ \mu \leq\ \eta \tag{8}$$

The character ρ can be any American English Phonetic Alphabet character as in Figure 16 or the SQL dependent wildcard character φ. The resultant search string after all conversions shall be contained by the array $O(j) = \{\rho_1, \rho_2, \dots, \rho_j\}$ that is the substitute string for the SQL query. SQL is a standard language for accessing database, but its implementation varies from language to language depending on the library used to implement databases access. Known Libraries such as LINQ or Dataset in C# vary their implementations for SQL database access. Special attention is necessary to ensure that each technology provides wildcard implementation of search pattern.

Then, the resultant pseudocode for the PDS search string is the following:

$If\ \eta \leq 5$; if length of the morpheme is less than 6 no transformation is used

{

$O(j) = W(i);$

$j = j + 1$; Increments the index for Output Array and points to the next character

$i = i + 1$; Increments the index for Input Array and points to the next character

}

$while\ i\ \leq\ \eta$; Morpheme is bigger or equal than 5 – requires analysis.

{ ; Multiple cases to be tested an wildcard set as needed

$case\ W(i) =$ "b" or "d" or "f" or "g" or "h" or "k" or "l" or
"m" or "p" or "r" or "v" or "w" or "y"; Checking for single phone followed by space

{

$i = i + 1$; Advance to next character in input array

$if\ W(i) =$ " " ; If it is a space copy to output previous, phone and space

{

$O(j-1) = W(i-1)$;

$O(j) = W(i)$;

$\mathrm{i} = \mathrm{i} + 1$;

$j = j + 1$;

$break$;

}

$while\ \ W(i) ==$ "b" or "d" or "f" or "g" or "h" or "k" or "l" or
"m" or "p" or "r" or "v" or "w" or "y";

{

$i = i + 1$; If a repeated character increment Input Array pointer until it does not repeat.

; No increment of Output pointer since no copy takes effect because we are skipping repeated characters.

}

$break$;

$case\ W(i) =$ "c" ; Checking phoneme “c” or “ch”.

```
            {
                    i = i + 1;
if W(i) = "h"; is the next phone a "h"?
                    {
                        O(j − 1) = W(i − 1); Then copy to output
                        O(j) = W(i);
                        i = i + 1; Increment pointers to look at next phone
                        j = j + 1; Ready for copy on empty slot
                        break;
                    }
while W(i) = h; checking for repeated h's and skipping their copy
                    {
                        i = i + 1;
                    }
                    break;
            }
case W(i) = "d"  ; checking for phone "dh"
            {
                    i = i + 1;
if W(i) = "h" or " "; Found and correct, copy to output array
                    {
```

$O(j-1) = W(i-1);$

$O(j) = W(i);$

$\mathrm{i} = \mathrm{i} + 1;$

$j = j + 1;$

$break;$

}

$while\ W(i) =$ "h" ; Check for repeated “h” and avoid copy to output

{

$i = i + 1;$

}

$break;$

}

$case\ W(i) =$ "e" ; check for correct phones “eh” “er” and “ey”

{

$i = i + 1;$

$if\ W(i) =$ "h" or "r" or "y";

{

$j = j + 2;$

$O(j-1) =$ "_"; If found insert wildcard character for each letter

$O(j) =$ "_";

$\mathrm{i} = \mathrm{i} + 1;$

```
            j = j + 1;
            break;
          }
while W(i) = "e" ; compensate for repeated "e"
          {
              i = i + 1; no copy to output
          }
         break;
     }
case W(i) = "i" ; check for combinations "ih" "ir" or "iy"
     {
         i = i + 1;
if W(i) = "h" or "r" or "y"; found then copy wildcard to output
         {
            j = j + 2;
            O(j - 1) = "_"; Replace previous with wildcard
            O(j) = "_"; Replace current with wildcard
            i = i + 1;
            j = j + 1;
            break;
          }
```

```
        while W(i) = "i"; repeated characters avoided.
        {
            i = i + 1;
        }
        break;
    }
case W(i) = "n" ; checking for correct phone “ng”
    {
        i = i + 1;
        if W(i) = " " or "g";
        {
O(j − 1) = W(i − 1); found copy it to output
            O(j) = W(i);
            i = i + 1;
            j = j + 1;
            break;
        }
while W(i) = "n"; compensate if repeated
        {
            i = i + 1; look again and do not copy to output if found
        }
```

```
            break;
        }
case W(i) = "o" ; check for “ow” and “oy”
        {
            i = i + 1;
            if W(i) = "w" or "y";
            {
                O(j − 1) = W(i − 1); Found! Copy to output
                O(j) = W(i);
                i = i + 1;
                j = j + 1;
                break;
            }
while W(i) = "o"; Found repeated character. Compensate and no copy to output.
            {
                i = i + 1;
            }
            break;
        }
case W(i) = "s" ; Looking for “sh”
        {
```

```
            i = i + 1;
            if W(i) = " " or "h" ;
            {
                O(j − 1) = W(i − 1); Found just copy. No wildcard here
                O(j) = W(i);
                i = i + 1;
                j = j + 1;
                break;
             }
             while W(i) = "s"; Check for repeats
             {
                     i = i + 1;
             }
            break;
       }
case W(i) = "t" ; checking for "th"
       {
            i = i + 1;
            if W(i) = " " or "h" ;
            {
O(j − 1) = W(i − 1); Found! Just Copy
```

$O(j) = W(i);$

$\mathrm{i} = \mathrm{i} + 1;$

$j = j + 1;$

$break;$

}

$while\ W(i) =$ "t"

{

$i = i + 1;$

}

$break;$

}

$case\ W(i) =$ "u" ; checking for "uh or "uw"

{

$i = i + 1;$

$if\ W(i) =$ "h" or "w";

{

$O(j-1) = W(i-1);$ Found it! Copy to output and increase pointers

$O(j) = W(i);$

```
                    i = i + 1;
                    j = j + 1;
                    break;
                  }
while W(i) = "u"; checking repeats
                  {
                        i = i + 1;
                  }
                break;
         }
case W(i) = "z"  ; Looking for “zh”
         {
                i = i + 1;
                if W(i) = " " or "h";
                {
                    O(j − 1) = W(i − 1);
                    O(j) = W(i);
                    i = i + 1;
                    j = j + 1;
                    break;
                  }
```

```
        while W(i) = "z"; Checking for repeats
        {
            i = i + 1;
        }
        break;
    }
case W(i) = "a" ; checking for "ae" or  "ax" or  "an" or  "aa"
    {
        i = i + 1;
        if W(i) = "e " or "x" or "n" or "a";
        {
        j  =  j + 2;
          O(j − 1) = "_";Inserting wildcard before and after current pointer
          O(j) = "_";
          i = i + 1;
          j = j + 1;
          break;
        }
        while W(i) = "e " or "x" or "n" or "a"; checking for repeats
        {
            i = i + 1;
```

```
            }
            break;
        }

}
```

As defined by the algorithm the, two arrays exist: *W (i)* that serves as a placeholder for the ASR, and the array *O (j)* that is the output for the search pattern utilized to perform a SQL query. The image on Figure 20 shows the contents of each array after execution. The confusion phonemes have been replaced by wildcard characters that permit a search of the string based on the prevailing phones after the wildcard substitution. It is important to recall that for phonetic words of length smaller than six characters an unmodified phonetic search pattern will be used instead.

Figure 20: PDS Example Input and Output Arrays

W(i)	p	r	eh	z	ih	d	ah	n	t	t
O(j)	p	r	__	z	__	d	__	n	t	

In our specific C# implementation using Datasets to communicate with the SQL server database the final search string for this example is:

%p r ** z d ** n t%; where 8 = to wildcard character "_"

The suggested stream in passed to the controlling database API for a search using SQL language LIKE clause. The results from the search are categorized by GUID and timestamp denominators. The original Dialog (Video) or Context GUID prevails after the search as a feature to trace back the origin of the found word with all its position and tracking information such as Utterance GUID and Dialog GUID. The results are compared automatically with the same search using the alternate WORD, PHONEME, and METAPHONE search algorithms.

CHAPTER FIVE: DOUBLE METAPHONE STRATEGY

The ability to search though data using phonetic information provides advantages that a standard word search cannot match. Phonetic matching can be used to find strings with similar pronunciation that sound alike regardless of their actual spelling. However phonetics search is not perfect, but in the context of categorizing video through its audio, the errors generated from the ASR conversion convert to a failed word search because the word sought might have lost its meaning in the translation, i.e. the word “Sea” was recognized as “See” and never found during a word syntax search. A phonetic search will find both words. Further, phonetic conversion analysis allows the creation of search algorithms as PDS.

Phonetic matching needs to be fast and accurate (Justin & Philip, 1996), we have seen though the result of this research that accuracy of the phonetic search is loosely tied to effectiveness. The goal is to not only provide a word matching capability, but also augment the retrieval with additional word content for the user to discern. Know algorithms are many, but SOUNDEX is where the Double Metaphone is born.

Soundex is a method for phonetic indexing patented by Robert C. Russell in 1918 (Black, 2007). His invention at the time was related to card or book indexing where the names where entered and grouped phonetically rather than alphabetically. Soundex provides a method to categorize or group names that have the same sound regardless of spelling. Search performance is maximized because only the group with the same sound will be searched. Soundex builds on the premise that the American English language has certain sounds that for the nucleus of the language and that are better represented by a phonetic representation rather than an alphabetical or syntactic approach. Each Soundex code has a letter followed by three numbers that describe the group to which the name belongs categorized by its sound, i.e. Washington is coded W252. More specifically, W for the first character, 2 for the S, 5 for the N, 2 for the G, and all other letters avoided. The development of a modified Double Metaphone Strategy is based on the evolution of Soundex. Soundex codes begin with the first letter of the surname followed by the three number code that represents the consonants that remain after the transformation. The coding avoids coding letters A, E, I, O, U, Y, H and W.

Table 5: Soundex Coding Scheme.

Soundex Coding		Codes
1	=	B P F V
2	=	C S G J K Q X Z
3	=	D T
4	=	L
5	=	M N
6	=	R

The letters A, E, I, O, U, Y, H and W are excluded form coding.

Double letter should be treated and single letters and Letters that have the same code number should be treated as one letter. Names with prefixes such as VanDausen code with and without the prefix, i.e. V-532 or D-250. If a vowel separates two consonants with the same code the second consonant is coded. Alternatively, if letters H and W separate two consonants the consonant to the left is coded.

The interesting fact of Soundex is that it can retrieve multiple syntactic expressions of the same sound. Thus, a feature that PDS does not consider. Database searches are often confronted with the problem of searching words in a large imperfect array of word. Often a search is done for a word that was misspelled or spelled in an unexpected way, never to be found by exact matches. In the context of Video information retrieval, we are more interested in approximate matches because the user discriminates at the end the accuracy of the search. The more related information that is available that describes the content of the transcribed audio file, the better the outcome of the search will be.

Soundex is not perfect since it does achieve high match throughput, some results may prove irrelevant. The biggest problem with Soundex is that after a determined length it does not continue to examine the word however performs its task of searching phonetically very efficiently. Studies done by a language analysis company, demonstrate that Soundex suffers from 11 deficiencies such as poor precision, sensitivity to noise and unranked returns within others (Patman & Shaefer, 2003) . A Before we forget, Soundex does not work with numbers.

Soundex evolves to American Soundex in 1930 to adapt to American names and further evolves to the Daich-Mokotoff Soundex in 1985 adapting to Easter European Names.

Most recently in the period from 1990 and 2000 Metaphone and Double Metaphone, versions of Soundex emerged. Metaphone generates the encoding based on how the name is pronounced instead of its spelling and works with the English language only. It is based on the entire name and not a subset of names. Double Metaphone created by Laurence Philips

(Lawrence, 2000), that produces two encodings for each name and included foreign pronunciations. The encoding is done using the initial part of the name.

The latest iteration of Soundex is the Beider-Morse system in 2008. The algorithm attempts to reduce the number of false positive matches by determining the language of the spelling and applying pronunciation rules to it.

On this dissertation we are interested in retrieving as much information as possible related to a words search or topic. It is of our interest to create additional matches that might be of interest of the user, some me be of the false negative type. To elaborate on the match type, matches that are found by the system are positive searches, while the unfound matches are negative. The positive matches that are relevant for the particular application are true positives while the irrelevant matches are false positives. Where the line is drawn regarding the inclusion of false positives and negatives is rather subjective since the user better knows about the relevance. Therefore, a method such as the Double Metaphone allows us to experiment with the false positive side of the available data and evaluate how it fairs. It is important to highlight that the Double Metaphone has not been used to search text in general as in our specific interpretation of Video ASR conversion. We demonstrate that with this method we are able to retrieve additional false positives that nether neither PDS nor phonetic search itself can accomplish. In some instances, the search shows an improvement of 1.32% to 250% over the baseline. It is not perfect since it cannot retrieve numbers, 0% difference from the baseline; it also increases the false negatives as the word letter count decreases below 4 with a -71.12% worse than the baseline. However, other methods explored do obtain the desired results.

Double Metaphone Implementation

The Double Metaphone as explained is an evolved Soundex. Our implementation of the double Metaphone required changes to the algorithm to accommodate all words and cross-reference. This implementation requires two phases, a reallocation of the corpus indexed for Metaphone search and a search counterpart that converts the sought word to each Metaphone and searches for the content based on the Metaphone information.

The Metaphone algorithm is an alternative to Soundex that is used to search phonetically for names in a repository that contains large lists of names and surnames; it has been reproduced from the original paper (Lawrence, 2000). The basic rules for Double Metaphone are based on the reduction of the names reduced to one for the following 16 letters:

[B, X, S, K, J, T, F, H L, M, N, P, R, 0, W, Y]

If the word begins with any of the following combinations, drop the first letter.

[“ae”, “gn”, “kn”, “pn”, “wr”]

If the beginning of the word is ‘x” change it to “s” and it begins with “wh” change it to “w”. Thereafter for each letter, if a “B” is found leave it as a “B” unless at the end of a word after “m”. Letter “C” transforms to:

[‘X” if “cia” found or “ch” found]

[“S” if “ci”, “ce” or “cy” is found]

[Ignored if “sci, “sce” or”scy” is found]

[Otherwise C, including “sch”]

Letter “D” transforms to:

[‘J” if “dge”, “dgy” “dgi” are found]

[“T” otherwise]

Letter “F” transform to “F”.

Letter “G” transforms to:

[Ignored if “gh” and not at the end or before a vowel]

[“G” if “gn”, or “gned” is found]

[“G” if “dge” is found]

[“J”, if before “i”, or “e”, or “y” if not double “gg”]

[Otherwise, “K”]

Letter “H” transforms to:

[Ignored if after a vowel and no vowel follows, or after “ch“, “sh”, “ph”, “th”, or “gh”].

[Otherwise H]

Letter “J” transforms to “J” regardless.

Letter K transforms to:

[Ignored if after “c”]

[“K” otherwise]

Letter “L” transforms to “L” regardless. Letter “M” transforms to “M” regardless and Letter “N” transforms to “N” regardless.

Letter “P” transforms to:

[“F” id before “h”]

[“P” otherwise]

Letter “Q” transforms to “K” and letter “R” transforms to “R” regardless.

Letter “S” transforms to:

[“X” if “s” is before “h” or if in “sio” or “sia”]

[“S” otherwise]

Letter “T” transforms to:

[“X” if in “tia” or “tio”]

[“0” if before “h”]

[Ignored if in “tch”]

[“T” otherwise]

Letter “V” transforms to “F” regardless.

Letter “W” transforms to:

[Ignored if not followed by a vowel]

[“W” if followed by a vowel]

Letter “X” transforms to KS regardless.

Letter “Y” transforms to:

[Ignored if not followed by a vowel]

[“Y” if followed by a vowel]

Letter “X” transforms to “S” regardless.

Notice that the code since it is designed for consonants it does not work with numbers. However, we will use it to encode anything contained in our baseline ASR conversions. The Double Metaphone creates a second Metaphone code to address the importation of names into the English Language from languages such as Slavic, Germanic, Celtic, Greek, French Italian Spanish and Chinese ("Double Metaphone," 2010).

Table 6: “PHONETIC_WORD_CODE” Table Sample

WORD GUID	CONVERSATION GUID	CONTEXT GUID	WORD	PHONETIC KEY1	PHONETIC KEY2	WORD POSITION	CHAR POSITION	TIME STAMP
ac748d11-2db5-4065-93fd-e1e04c2dfe43	2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F	27018EDD-F6D4-4829-A58E-7A1F134174AC	out	AT	NULL	44	245	5/14/2010 19:14:25
adade126-7258-42ec-969a-1fa340adf5e7	2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F	27018EDD-F6D4-4829-A58E-7A1F134174AC	background	PKKR	NULL	28	149	5/14/2010 19:14:25
b256d814-63a4-4806-affe-ad95e943272b	2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F	27018EDD-F6D4-4829-A58E-7A1F134174AC	financial	FNNS	FNNX	60	337	5/14/2010 19:14:25
b2695f85-65c9-421f-89a1-85480ac5d543	2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F	27018EDD-F6D4-4829-A58E-7A1F134174AC	opt	APT	NULL	43	241	5/14/2010 19:14:25
b55a15f0-6e25-455a-9a7d-d2e9a9a6062c	2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F	27018EDD-F6D4-4829-A58E-7A1F134174AC	the	0	T	37	197	5/14/2010 19:14:25
ba4fc4c0-9b55-44ff-8b19-a171f6901c1a	2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F	27018EDD-F6D4-4829-A58E-7A1F134174AC	that	0T	TT	16	81	5/14/2010 19:14:25

The reallocation data and the indexing required for Metaphone use required several changes to the original system design. Dual tables were created to host the Metaphone information separately for the video context and the video translation since the entire corpus had to be restructured and used with Metaphones. The tables are

"PHONETIC_WORDS_CODES_CTX" and "PHONETIC_WORD_CODE" each with similar data as in the sample Table 6. The function to convert all the video collected to Metaphone. A significant contribution under this process can be noticed; under this schema, we can provide word-tracking information a feature that the Metaphone algorithm does not provide. This tracking information is available because our initial ASR conversion schema and indexing which allows us to track each word back to its original position as in the original video. Further, each word can be associated by reference to the original utterance within the video, features that allow using this schema for video search. The possibility of providing time information tied to position allows a GUI developer to provide a search interface where the video can be self-paced by with the help of a slide bar or control knob. The slide bar's origin is the beginning if the video and position 0, and the end position e.g. 38min and 23sec. as the slide bar is moved the video advances by utterance which shows the words from on each utterance while the user fast forwards the video to a self asserted relevance within the video.

The application that indexes the current corpus reads the corpus in by word and transforms each word the equivalent Metaphone and additional reference information into the table similar to Table 6.

The second application considered the search interface performs the search through the database by transforming the search word to its equivalent Metaphones and searches for the equivalent token in the Metaphone database tables retuning all matches found. Experimentation was done with the length of the Metaphone that can be varied, however four characters works best for the content analyzed. As evaluated in Figure 17, less than 2% of the content is bigger than 10 characters. Since this implementation of Metaphones is not on names and surnames only, perhaps it addresses all possible worlds within the context, four characters work best for the Metaphone length. Larger Metaphone length transformations increase the false positives. Further work can be done with Double Metaphone adaptation to be optimized for general topic search; however, such experimentation will require a separate ontology conversions for each Metaphone key code length.

The next chapter explains in detail the design and architecture of the proposed systems. The fist system is dedicated to the extraction of phonetic information, the indexing of data and, the context term retrieval; are all executed before search processes. As the data is organized, four types of search are juxtapositioned; baseline word search, phonetic search, PDS search and Metaphone. A separate architecture implements each search in a single interface that stores all the results for analysis.

CHAPTER SIX: PROTOTYPE SYSTEM DESIGN

This particular system implementation is composed of four components. Each individual component is part of a sequential pipeline can handle multiple jobs at once. Different versions of video indexing systems have evolved in an attempt to solve the problem on how to index large video automatically.

Our particular scenario has similar goals, however with different behavior. It is a fact that we do not have the resources to integrate our video indexing approach into an enterprise application; however possible since the audio striping an ASR translation, the context term retrieval and DB indexation and storage, are all done separately to maximize the use of resources. We dedicate our efforts in providing a different approach in translating and indexing audio utterances, where the phoneme extraction is preferred to text, perhaps it preserves somewhat the original sounds of the audio through phonemes that provide a different alternative in ASR transcript reconstruction and indexing.

With modest resources, it is possible to individualize each component for a multi-computer multithreaded environment to process a constant stream of video. An individual system encodes the video to audio using 16 bits mono channel and 16 KHz sampling rate. A separate system reads the available audio and processes it by a multithreaded parallel operating LVCSR with a scalable capacity to process multiple utterances at the same time from sections of multiple videos, which will be grouped synchronously in the database to be search by the proposed methods. A client interface can be made web-based as many browsers using web technology that will allow the user to search for videos. The web-client will retrieve the video context term information and video name associated with the content. The user thorough the web-client will be able to select the video of interest and further skim though the video by utterance, while the words used for the search are displayed in the timeline of the video for reference. At this point, the user can search within the video for words of interest; the interface will return the matches available on the video and post on the video time line its location for reference. Multiple clients can be querying the farm databases that interact with the client through phonetic data, and transfer pieces (utterance groups) of video at a time as the user makes his selection. However, our test bed is regrettably based out of a distant budget using COTS software and hardware; nevertheless, it demonstrates the feasibility and power of such large implementation. It juxtapositions methods not used by databases today such as approximate search exposed in this work.

As explained earlier, two separate applications are integrated to support the concept of a video indexing and search environment using phonetic retrieval. The first application extracts phonetic information from the original videos, as well as word information. Aligns words and morphemes by location and stores it into a DB. I further analyses the data and retrieves the context based with the help of external term summarization API's from Calais and Yahoo and stores the information related to its source with the help of an Aho-Corasic pattern matcher algorithm. Meanwhile, it also provides a data structure for WER calculations and transcripts performed in a Virtual Linux Environment with the help Virtual Box ("Download Oracle VM VirtualBox," 2010).

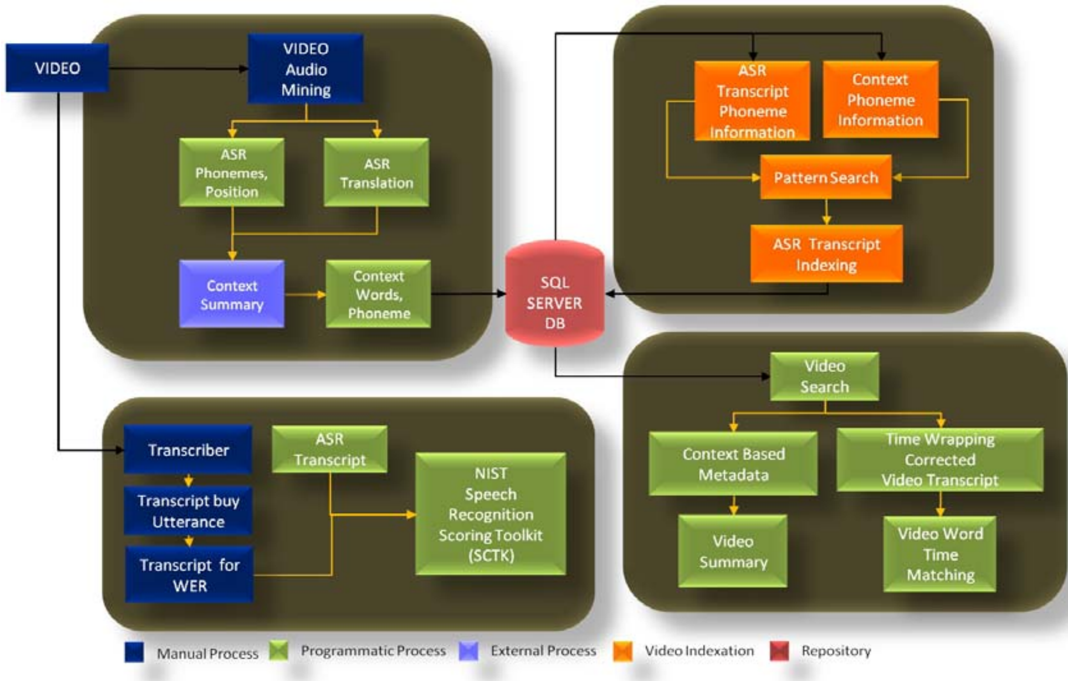

Figure 21: Phoneme Based Video Extraction Architecture

Figure 21 illustrates the general architecture used for the Phonetic extraction and Database Indexing Utility (PEDIU) as conceived initially. The first of the four modules shown contains the Video conversion, ASR translation, and Context Term Extraction (top left corner). The second module (Bottom left corner) is where the transcription of the audio is performed manually and used by NIST toolkit (NIST, 2010) to calculate WER. The third module (top right corner), represents the Aho Corasick pattern matching algorithm that locates Context Term information within the ASR translated text and retrieve indexing information to map the new context information with the video content. Finally, the fourth module represents the word spotting phonetic search that allows searching through video context and video content information, and further storage of all information for analysis.

For our test bed, we selected video files from the news weekend program "Meet The Press" simply because the diction of the interview is excellent, cross talk is avoided, and the transcripts to the selected videos were provided. We captured about 5 hours of video files divided in groups thirty minutes and one hour in length sampled at 16 kHz mono 16 bit encoding. Matching translations were cleaned from artifacts as well as the video was matched with the translation by hand. Background noise due to advertizing or clutter was left intact if also located in the manual transcript, otherwise extracted from the original audio sample using DSP.

The audio extracted from the video converts its content into wave files that are stored as part of the file system. This stage is a pre-processing stage that can be done asynchronously. The audio output generated can be selected by the user or processed automatically and converted concurrently into a phoneme and word representation by the ASR engine. The content is indexed and time stamped and placed into the repository. It is important to discern that each audio conversion is an untrue translation of the original audio due to the ASR inherent translation errors. Thus, two identical videos do not produce identical ASR translated copies; therefore, multiple copies of the same video are included as part of the test set to induce variance due the ASR errors. However the errors induced in the word ASR conversion, are avoided in the phonetic counterpart somewhat as phonemes, because the phoneme preserve the sound of the word. Nevertheless, phonetic translation also suffers from errors because of poor diction, ambient noise and second and third sound arrivals of the recording sound, are within other aspects the inaccuracies of recorded audio. Additional errors also are generated due to OOV words that do not exist in the trained vocabulary that the ASR generates its best guess according to ASR defined HMM. Consequently, some audio to phoneme translations are correct, but on a similar comparison, audio transformed into words using ASR dictation can generate increased OOV errors. By using phoneme conversions directly, we can reduce out-of-dictionary word errors since the original semantics of the word is preserved through phonemes.

To extract the content of the ASR translated document we use a clever *context finder* implementation with the help of term extraction API's provided by Calais and Yahoo, many others are available. It uses the ASR input text and summarizes its content by providing context related information aided by web search. Context related words are extracted from the original ASR translation and later converted to phonemes using the SAPI synthesizer included in Windows 7 operating system (controlled by SAPI) that will become part of the reconstruction of the original ASR document. The application collects the documents stored in the database and extracts a word summary that is converted in to phonemes generating an equivalent phoneme based string array that represents the summarization of each ASR document.

Interestingly, for every phoneme-word extracted by the summarizer there is at least a one to one correspondence with the original video, however it is typical to find one-to-many matches because a context word can be found repetitively along the original ASR document.

The *Phoneme Pattern Matcher* used to create the relation of context to original video, is based on the genome pattern matching algorithm Aho-Corasick (Aho & Corasick, 1975). The Phoneme Pattern Matcher collects information from the database regarding the original ASR phoneme translation and the context information to find each phoneme and return its position. All this is done swiftly as the algorithm is capable of searching for many patterns in one pass through the document. Most other pattern algorithms search for a word at a time based on a variable windowing method that scans through the entire document one pattern at a time. This matching is done rather quickly, for all hour-long videos the matching of over 100 words is done in less than 500ms.

The block diagram on Figure 22 depicts the current architecture in detail for the Phoneme Based Video Extraction; our system implementation transfers data through a pipeline that carries information from client utilities to the contextualizing algorithm. In a multi-party dialog system, users may interact with several client-side tools or automate the process entirely. It may be of the interest of web development search parties to scan through the content of videos at a usage dip and summarize the contents of each Video at real-time. By including summaries of lengthy videos as part of a metadata description, user search can be done effectively without opening the video content or describing the video manually as part of the description metadata. With the proposed system, this can be done automatically.

The detailed phoneme and text translation used by Phone Based Video indexing and summarization can be localized in the top center section. Detail information regarding the files used for the different phoneme and word conversions and their interaction with the ASR and database is also shown for the reader’s consideration.

Nevertheless, the loss of conversational information, results from performance limitations or data filtering. Therefore, some words will be lost in translation due to factors external to an ASR conversion.

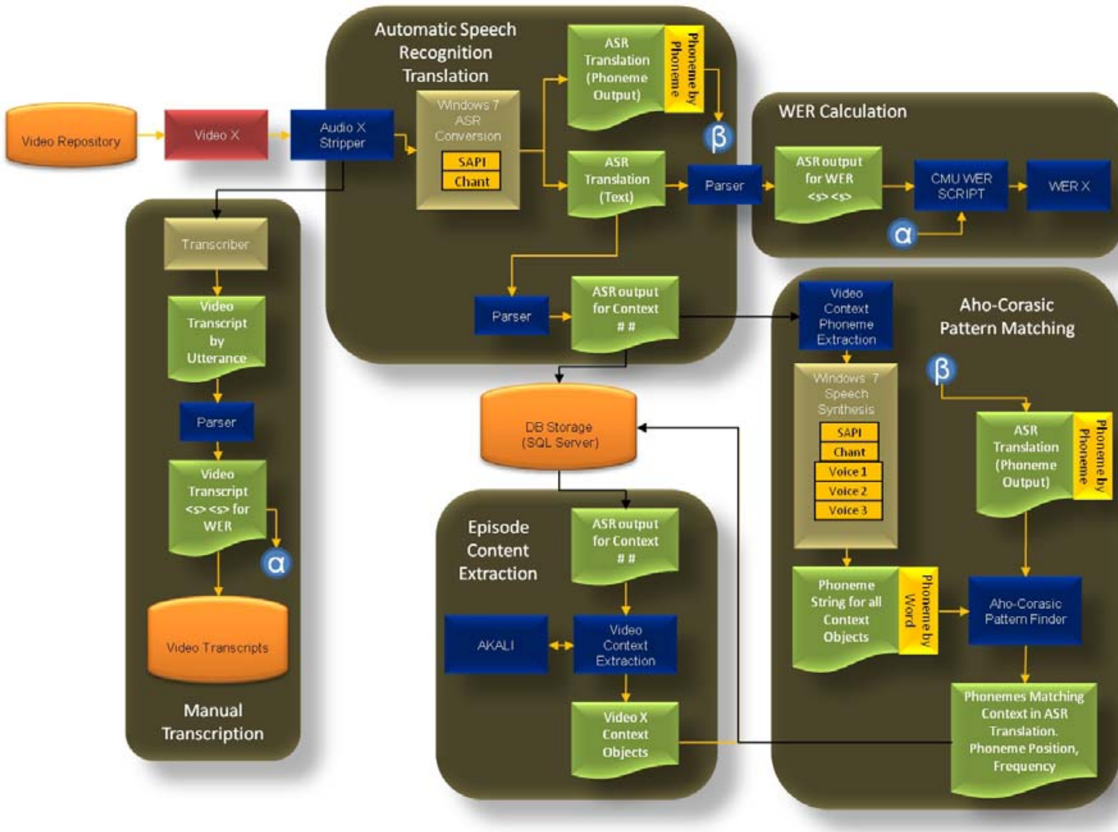

Figure 22: Phoneme Based Video Indexing

A common example of this can be observed in the speech recognition modules such as the one used on this system. Speech recognition in unrestricted domains can be subject to varying degrees of accuracy that erodes topic coherence by introducing noise (Gurevych, Malaka, & Porzel, 2003). In addition, the module itself may enforce restrictions on the data it returns. Therefore exact translations of the videos are mostly succesful as references to the original document than as trascripts. For the efect of video search and sumarization the results are encouraging. World wide web sites alocating video can preporcess millions of videos and describe them automaticaly without user intervention.

The contextual extraction tool has been tested with great results on other research systems. Some example of these include distributed speech recognition modules for each user; avatar or agent representative tools (DeMara et al., 2008); an interpreter or dialog manager; and multimedia presenters in the form of dashboards, interactive tables, etc. All of these subsystems represent possible points of origin for the focus of a conversation. Since these may be developed independently of each other, they are usually integrated in an ad-hoc manner and suffer from information constraints (Le Bigot et al., 2007).

As part of the indexation process, we add to the original ASR transcription contextual summarization that can be used to describe the content at a higher level such as metadata does describe objects in WEB searches. Thus, we address the tasks of storage and retrieval of the spoken dialog system. More importantly, we describe two contributions: (1) a process for determining the prevalent contexts of the current dialog composed of utterances, and (2) a prototype system for accomplishing the aforementioned tasks. For the purposes of this discussion, we will focus on a broad, finite domain of dynamic contexts obtained from video. Within this scope, we refer to a *conversational context* as the set of topics suggested by the utterances of all parties involved in the dialog. Moreover, we specify a *dynamic context* to be an abstract construct with a predefined structure, but whose possible range of attributes are not known a priori. The context term retrieval is composed of three components implemented in our gisting architecture. These three modules consist of database interfaces, a back-end database, a contextualization process or API, and several analysis services. We define the general purpose of these components in the proceeding paragraphs.

Through the database interfaces, the architecture services requests for recalling events that have been contextualized and stored in a database by the Calais and Yahoo API's. Our implementation of database memory interfaces is loosely coupled. Weick (1976) first introduced loose coupling as a design pattern in which the knowledge of one class with respect to another on which it depends is limited to include only the interfaces through which they interact. In our case, the loosely coupled interfaces hide the implementation of processes internal to the database architecture from the audio/video indexing systems that might use it to store or retrieve content. At the same time, they allow communication to occur between the memory architecture and systems that use it.

A back-end database running on a server forms a crucial part of our gisting architecture in that it serves as the storage medium for context and ASR translation data and the internal processes that manipulate conversational information. In addition, server-side processing allows us to remove the data-intensive operations of contextualization from machines that may already be taxed while transcribing conversations. By following such an approach, we ensure minimal side effects on the real-time operations of the indexing systems.

The third component of the architecture, a contextualization process, is responsible for managing the input of interaction data, storing Term extraction it in the database memory, indexing the utterances, and deciding which utterances are relevant for a query request. It exploits custom storage structures to store, index, and retrieve episodes.

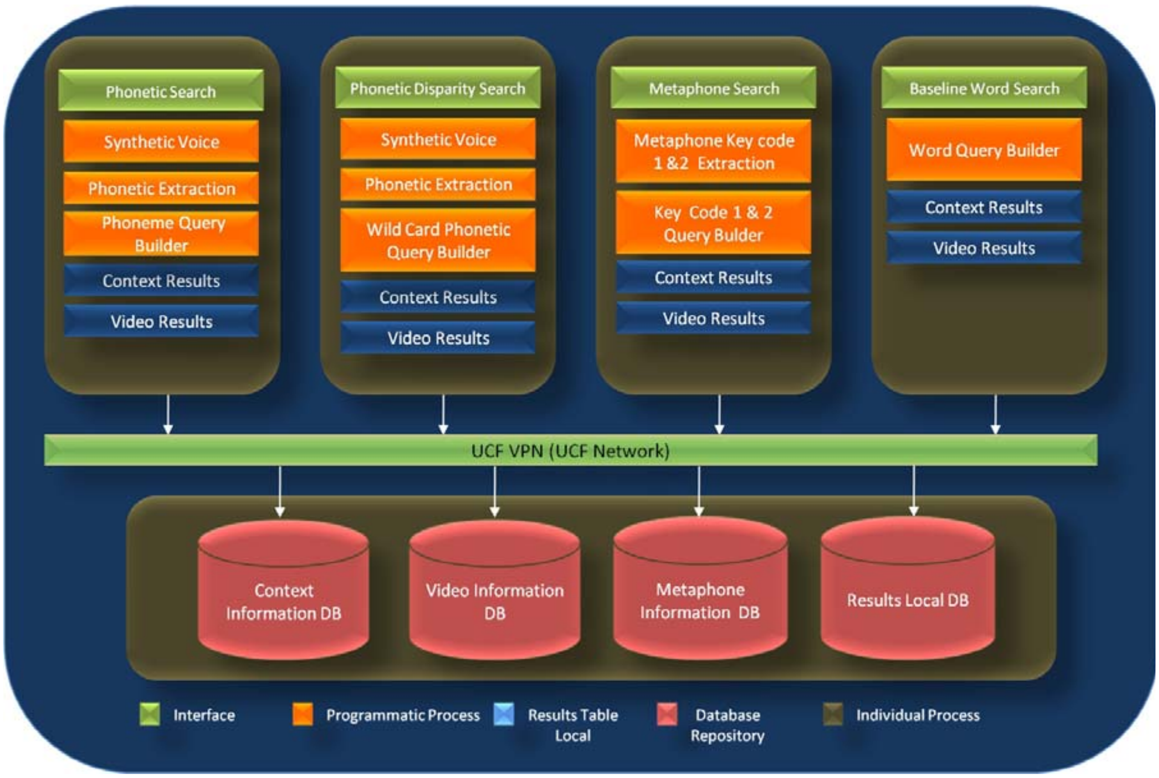

Figure 23: Phonetic, PDS, Metaphone, and Word Search Architecture

The implementation if the interface that gives access to contextual, translated and Metaphone converted information can be studied in Figure 23. Four individual systems perform loosely coupled search components. Each of the four systems has a query component used to extract requests of data from the memory database that contain contextual term information and video content information. The contextual information is a dynamic context that was created using external services that argument the original video content based on web information. The conversational content is the data obtained from the video collection as transcribed by the ASR. Dynamic and conversational context is available in separate databases. The context database stores the context terms retrieved using the external API's while the Video information database stores all the video ASR translated data for both phoneme and word tokens as a result of the ASR translation. The Metaphone database is a duplication of all the video content data where each word within an utterance has been converted to its Metaphone equivalent, therefore creating two keys that describe each word phonetically.

Phonetic Search

The phonetic search system performs a standard phonetic search by transferring the sought word into its phonetic interpretation aided by a voice synthesizer and phonetic extraction tools. Programmatically, the word search is converted to its phoneme components and used to build a phonetic query that will retrieve all identical matches using the described phones.

It is known that this particular search will underperform, but was created for comparison purposes. Vocabulary independent searches are known to suffer from high word error rates , the phonetic lattices generated that match spontaneous speech are known to score high in errors (Seide, Peng, Chengyuan, & Chang, 2004). Since the phonetic search only looks for exact phonetic matches, it will miss any word that does not match exactly. It is of our interest to perform an approximate of variable search to retrieve all sound related matches possible since an extract phonetic search will not include variations of the phoneme sought. As the query is processed result regarding the success of the search is retrieved it is stored in remote results database. The databases results are also duplicated in the interface as tables showing the matches for contextual and video content. In other words, dynamic and conversational content is available to the phonetic search based on a standard query that searches the morpheme translated by the synthetic voice. If the morpheme is found in the phonetic stream, it is retrieved with the related utterance and video information.

Phonetic Disparity Search (PDS)

The second search system is the Phonetic Disparity Search, is a variation of the phonetic search. It is much different since it provides four general features.

The first feature is a voice synthesis translation to morphemes that express the search. The second feature is housekeeping and cleaning operation that mitigates errors induced by the ASR conversion.

The third feature creates a query that is similar to a phonetic wild card search, however using heuristics based on the observation of different synthetic voices and ASR translations. The rules are built into the system; they preprocess the query to build an approximate search. It is of our interest to receive as much false positives as possible.

The fourth feature is for analysis it creates juxtaposition with the other search methods. As the data is retrieved, information regarding the origin of the data, frequency of the words found and if the retuned data belongs to the conversational context or the dynamic context.

In brief, the PDS search will take a word in its dialog box convert it to a morpheme by a synthetic voice and create a query based in a rule set that determines what is the best wild card formulation for the search. Then, performs the search and retrieves the relevant information from the context and video content databases while it presents numerical data that describes the search. The results of the search are shown on a table and included as a tab on the interface. Figure 23 illustrates the interaction between the systems, particularly PDS will search through the context and video content databases individually as its results are shown separately also.

Double Metaphone Strategy Search

The Double Metaphone Strategy is a very fast way to implement an approximate search. This particular search used its own database schema created by a previous and separate indexing step. The Double Metaphone search has four functions.

The first function is to provide an interface to the search where the user can type the decided word-spotting token.

The second function relates to the extraction of the two Metaphone key codes from the search box content. A Metaphone function extracts the key following the Metaphone algorithm described in Chapter 5 passes the value to the query builder and performs the search on the Metaphone Information Database as described in Figure 23. The Double Metaphone Search contains all the necessary logon information to access the external database; however, a VPN access client is necessary to grant access to the UCF network.

Returned matches are posted within the interface in two tables that contain context term retrieval information and video content data for analysis.

CHAPTER SEVEN: RESULTS AND ANALYSIS

Our objective is to demonstrate that the performance of the proposed algorithm can significantly minimize the effort required to categorize and index video. Furthermore, a phoneme disparity search provides improved matching capabilities when compared with its counterpart word baseline search. The phoneme indexing and search combination can be used in many applications requiring the storage and auto indexation of video information. Typical examples are web-based video search dialogs, categorization of news casting video libraries, video recorded courses etc. The Metaphone search extends the search capabilities of PDS by locating OOV of interest, by increasing the false positive matches for the sought word showing a 250% percent increase in positive matches over the word baseline.

Before we present the results of this study, we briefly explain the mechanics involved in this two phased process. Initially, the ASR translation, context retrieval and indexing is performed, while later the different phoneme search methods are used to demonstrate text occurrence and matching. Five hours of ASR, converted video is used as test material. Phonetic lattices have been generated as well as word lattices that represent the content of the video samples, aided by the ASR contained in Windows 7 operating system. The content has been distributed in three databases that support the different schemas necessary to support each search algorithm.

Our baseline word search performs a word spotting within all the corpus of broadcast news and context term data. The results are compared and evaluated to determine which method provides better results as the outcome of an approximate search. We define approximate search as the query that not only retrieves the sought word, but words that have a similar meaning during a word spotting operation. We consider that because a phonetic translation preserves the sound of the each word in an utterance, it should be able to retrieve additional content not possible when searched by word or text.

Word error rates are used for comparison. It is known fact that vocabulary independent spontaneous speech ASR translations carry a high WER. The tainted ASR conversions from less than ideal audio material corrupt the oncoming ASR translation adding OOV words and making insertions and deletions of letters that corrupt the content varying the WER (Cardillo, Clements, & Miller, 2002). The video content file shown below which is also listed in the Appendix A, is 58 minutes long and has an error rate of 55.72 %. This video is a copy of broadcast news; it contains the voices of male and female in a sustaining dialog about politics and the economy.

ASROutputpaulson+greenspanclean.txt

TOTAL Words: 3286 Correct: 1831 Errors: 1685

TOTAL Percent correct = 55.72% Error = 51.28% Accuracy = 48.72%

TOTAL Insertions: 230 Deletions: 316 Substitutions: 1139

We are not too concern about the word error rate because our objective is not to improve it value, but use the extracted dirty information to extract video data to perform out test. However, the ASR has been trained with material related to the topics of conversation to provide a trained ontology for the ASR to do its job. Form the test perform this document presented the highest WER, however the average WER of all the videos is 48.1%. As mentioned, the material recorded has a direct impact in the WER. Word error Rates of 18% are possible under ideal conditions (Chelba, Silva, & Acero, 2007).

On this dissertation, we explore the capability of a phonetic search compared to our baseline word sear during word spotting operations. Phonetic searches inherit qualities that potentially provide better results. It is potentially accurate and fast, it is able to search an open vocabulary, it carries a low penalty for new words and fairs better with inexact spelling. We demonstrate that even tough error rates are perhaps on the prominent side, a PDS or Metaphone search will provide better results consistently within a few exceptions.

ASR Conversion, Contextual Retrieval, Indexation.

For our tests, we collected a small library of 30 to 60 minute news casting videos. Each video is paired with a manually generated human translation, later used as a baseline for comparison purposes. All videos used in the experiment contain newscast panel interviews of one or more personalities that address a specific topic e.g. economy, healthcare, and others. We noticed that each interview includes minimal crosstalk between the participants, however minimized by the newscast agency; yet included as part of our samples and our transcripts. However, the transcripts do not have information regarding each speaker; we simply preserve the utterances as created through the ASR as well as the speech. No pre-filtering or modification was done to the audio content, except for the omission of advertisement media to minimize out of content vocabulary indexing.

The video content was recorded directly from the web-based repository and stored for immediate audio stripping using a 16-bit 16 KHz sampling mono encoding, typical in voice recognition experiments. This process is manual in our experiments, but for large media libraries can be made in batches, automatic and concurrent (Alberti et al., 2009). Each time we capture the audio, the ASR conversion takes place using Windows 7 standard SAPI with default training that though custom software the word and phoneme translations are mined.

As a separate process, after the ASR translation is completed, the centralized memory contextualization algorithm parses the indexed text per video content and retrieves the context of the dialog, further storing contextual information into an external database. Next, the conversational topics are mapped onto the original ASR transcript for later reference by using Aho-Corasic pattern matching algorithm (Aho & Corasick, 1975). By design, the algorithm works best with large files, because it is catered for genome sequence pattern matching it can search multiple patterns in a single pass with amazing speed. It has proven to find about 100 contextual matches in 4 gigabytes of content in less the 500 milliseconds on our test system. It is used to match context terms with the ASR translation. Thus, it is possible to index large audio file content with the contextual terms and their locations within the ASR translation.

After all the video has been converted by the ASR, the transcripts built and the contextual terms extracted for each video, we can leave behind the first application and evaluate the Multi-search Application. This application serves as a client interface to all the database content and is a host for all the proposed search algorithms. The UI performs four search operations in a single pass; a phonetic search, a PDS and Metaphone search and the base line word search, and stores all the resultant data for analysis back into the repository. Let us begin by summarizing each search operation before we provide comparative results.

PDS Phonetic Disparity Search

In brief, PDS heuristics are based in the different combinations of phonemes to augment a single search based on the assumption that a mix of speech recognizers will corrupt the search. By using different speech vendors, each word query is then synthesized by different voices, each generating their own phonetic conversion. When each phoneme conversion is compared, the most probable errors are replaced by wild cards within the search improving the phoneme search dramatically. The extra characters inserted randomly due to ASR and TTS conversions, are also avoided. We are able to generate results up to 250% better than the baseline counterpart and consistently prove better than the baseline.

The ASR translation and categorization provides us with two sets of data for experimentation: the ASR translation and the conversational contextual words extracted from the ASR translation. Thus, the ASR translation is automatically converted to American English words and American English phonemes and stored in the database. The contextual word information is also stored in the repository in word and phoneme versions also. All stored material in the process, inherits OOV words from inaccurate conversion due to the imperfect source material and inaccuracies of the speech recognizer. We hypothesize that because the phonemes preserve the original sound of the word we can use phonetic information to expand the search further and hit related content where a word query would have failed due to a semantic loss at conversion.

Initially, we test the capacity of a distinct phoneme search versus a distinct word search using the smaller sized contextual data. The resultant phonetic search matches were less than the word search, a characteristic of the phonetic search because a single word can be represented by different morphemes making a general search difficult. Performing the same test on the original video content ASR data set produced similar results. However, as we looked for highly frequent words found by the search within the corpus and its related phoneme counterpart, we found a disparity within the word and phoneme search.

After evaluating the results of phonetic search on both data sets, we can infer that most of the discrepancy found was related to the synthetic voice used to convert voice to phonemes automatically. Although we were using Windows 7 SAPI for speech recognition and voice synthesis, the phoneme conversions from each system were from time to time different.

Table 7 shows the results obtained using different words with a modified search algorithm that uses three voice phoneme conversion and comparison for disparity, in addition to elimination of initial phoneme and elimination of phonetic characters based on the RMS value of the length of the original morpheme string.

Further analysis of the data demonstrated that the word search was finding not only matches for the specific test word, but also words that contained the root of the sought word, a task that the phoneme counterpart omitted. Detail analysis of the ASR translated data, revealed that the phoneme set had the correct phonetic information to describe the word sought phonetically, but its phonetic translation within the search application had errors caused by the speech synthesis; the phonemes used to construct an utterance were phonetically accurate but syntactically erred, therefore inserting unwanted allophones. For example, for the word "economy" the ASR to phonemes conversion sometimes generated the following phoneme.

Table 7: Search Test results based on Phonetic Disparity Search

Test Word Search	Hits Word Search	Hits Phonemes	Hits w/ PDS	Morpheme (1st voice)	Transformed Morpheme
Economic	23	20	69	Iy k ax n aa m ih k	% iy k ax n%
Economy	89	58	87	ih k aa n ax m iy	%ih k aa%
Jobs	84	84	163	Jh aa b z	%jh aa b%
Greenspan	20	20	24	g r iy n s p ae n	%g r iy n%
Harvard	8	4	59	H aa r v ax r d	%h aa r v%
President	133	3	189	p r eh z ax d ax n t	%p r e z%
Defecit	0	0	36	d eh f ax s ax t	%d eh f ax%
Deficit	26	28	36	d eh f ax s ih t	%d eh f ax%
Clinton	19	0	25	k l ih n t ax n	%k l ih%

ih k aa h ah m iy

Similarly, the speech synthesis engine used for the phoneme search client generated the following phoneme when the word "economy" was used.

ih k aa n ax m iy

Interesting is the fact that both conversions produce an accurate phonetic interpretation, but the second is different at the fifth phoneme. On the current phonetic algorithm, only the phoneme that matches the ASR phoneme conversion phoneme will become a hit. This result supports our hypothesis and our belief that there is not a one-to-one correspondence between a single phoneme expression and an ASR conversion of a word. We thrive in producing hits that the word search cannot perform due its relation to syntax rather than semantics, without deteriorating Precision and Recall.

Perplexed by the discrepancy, initially we decided to experiment with different synthetic voices. Surprisingly, we find that other vendor voices from time to time convert a word to phoneme differently; an event that suggests which phoneme is part of an incorrect phonetic translation or allophone. Sometime these can be different depending on who made the engine or if the followed the SAPI standard.

To compensate for allophones programmatically, we initially tried phonetic conversions using different languages within a single Commercial-Off-The-Self (COT) speech synthesis voice. The resultant foreign language phoneme conversion, replaced some out of language phonemes by empty spaces. As we tried different American English voice synthesizers, we noticed that in our test words, the phoneme differences were evident in their phonetic conversions. We found that there was an inherent phonetic disparity related to the different voices, but that such nuance affected the same phoneme or phonemes. Furthermore, in most test cases all synthesizers used converted words to phonemes identically.

We made changes to the algorithm to replace the common errors obtained from the different voices by a wild card within the search for the specific phoneme, the results proved better than baseline word search by 15.1% on high frequency words. Additional inspection revealed that for a range of input test words, each synthetic voice produced the same phoneme conversion output for repeated conversions or the same words except a few random words with extra characters. Nevertheless, each of speech synthesis voices will induce phoneme conversion errors that proved identical every time the same test words were used. Perhaps, allowing us to incorporate the differences in a search algorithm that would define PDS.

Double Metaphone Indexation

The Double Metaphone is a very powerful search algorithm specifically when the interest is of a approximate search and false positive returns. Its implementation requires a pre-indexing operation and an algorithmic implementation of key codes that groups similar sounding words together. The utilities created to index the entire corpus for context terms and translated video works efficiently without the use of multi-threaded applications or parallelism. The translated video corpus converted into 94541 records each with two Metaphone keys, however not all records have Double Metaphones key codes.

The utility converted 5 hours of text and phonemes obtained from the original news broadcast video in 47 seconds. The context term part of the database converted in less than 5 sec. Both of these conversions time are considered efficient and variable considering that all conversions are done onto remote databases using VPN connections.

As mention earlier, the conversion and indexation process is necessary for the use of Double Metaphone, because it generates all the Metaphone keys for every word and symbol separated by spaces. As we search for context term information or video translation data, the key codes are the tokens searched by a query. This is a very efficient process since it only looks at the records that contain matching tokens. For comparison, our baseline search has to look at the entire database to find a match, only optimized by the database engine. Although the wait times for any search on this system vary from less than 0.5 seconds to 3 seconds word queries shall take longer than any other search methods presented. Likewise, the corpus can potentially increase to thousands of hours of video; the key for speed is the indexation of the context terms with the original content. A search through the context terms will be many times faster than a search through the entire video library. As a video is selected, the user shall initiate a second level search to analyze and view the video. This process is much less expensive, and can be optimized to deliver groups of utterances at the time and cached on the client search terminal.

Multi-Search Results vs. Baseline.

We know present results of the investigation using a comparative approach. Our baseline measurements are performed based on a basic word query as we attempt to obtain a perfect match. We define *perfect match* as the query that return all the matches possible for a certain token in a word spotting operation. The exact relevant word has to be retrieved as many times as it appears in the corpus excluding non-relevant words. We concentrate on a word spotting, where we attempt to find and retrieve all the instances of a query token or word (Arnon et al., 2001).

All search operations are done as a group. Then, if a search is done for word XYZ, the UI will generate four searches. Although each search can be done in any order, we categorize in order across all results, the baseline word search first; followed by the Phonetic, PDS and Metaphone searches. As a result, each word, morpheme, or key code sought has a corresponding word that initiated the process and spawned to do baseline or phonetic queries based on the algorithms presented earlier on this document.

We evaluate the ability of a search to perform a perfect search. Moreover, we enhance the perfect search with an adaptive search that goes beyond an exact match. We propose that for web-based applications that manage a large library of videos, we do not have to find 100% of the words located in the video library corpus, but just a descriptive subset. The reader may question our assumption. The fact is that we cannot guarantee that all the words inside the corpus will match the original video, because the ASR translation induces errors that change the original syntax of the words.

We know from research that Word Error Rates for Broadcast news varies from low 18% to less than 60%, depending on the source material, ASR training and using the latest algorithms for ASR processing. However, the goal is not to perfectly transcribe a video, perhaps we aim to find its context terms partially in a library of videos as a discerning and selecting factor. Experimental evidence exists that exploring ways to retrieve small parts of the original content proves beneficial (Chelba et al., 2007).

The essential problem is that as a word is translated by an ASR, the translation occurs with errors at the output for many reasons. Inaccurate recordings, background noise, untrained

ASR, or incorrect pronunciation, all significantly contribute to the corrosion of the WER, and further deteriorate the phonetic and text translations. Even though phonemes tend to preserve the sound of the word, it is not 100% dependable.

Table 8: Juxtaposition of Baseline and Phonetic Searches

	Search Token	Precision Corpus	Recall Corpus	Precision Context	Recall Context
Word (A)	President	97.04%	94.25%	100.00%	79.17%
Phoneme (A)	p r eh z ax d ax n t	0.00%	0.00%	100.00%	16.67%
PDS (A)	%p r eh z __ d __%	100.00%	100.00%	29.17%	29.17%
Meta (A)	PRST	94.82%	100.00%	100.00%	100.00%
Word (B)	mortgages	100.00%	28.57%	0.00%	0.00%
Phoneme (B)	m ao r g ih jh ih z	0.00%	42.86%	0.00%	0.00%
PDS (B)	%m ao r g __ jh i%	100.00%	42.86%	0.00%	0.00%
Meta (B)	MRTK	100.00%	100.00%	0.00%	0.00%
Word (C)	Phd	0.00%	0.00%	0.00%	0.00%
Phoneme (C)	p iy ey ch d iy	100.00%	100.00%	0.00%	0.00%
PDS (C)	%p iy ey ch d%	100.00%	100.00%	0.00%	0.00%
Meta (C)	FT	0.00%	0.00%	0.00%	0.00%

We then analyze our search algorithms as we juxtaposed with the word baseline search by using an experimental bag of words (Hanna, 2006), some words are known to be in the ASR and others are unknown. Our tests demonstrate that in situations where the ASR translation is corrupted due to intrinsic conversion errors, phonetic searches can provide additional insight and improved retrieval results.

Table 8 is a comparison sample of results while testing phonetic searches against words queries for different word lengths. Earlier we defined three classes (A, B & C) for the length of a word based on it letters. Class A are words that vary in length from 6 to 13 letters, Class B varies between 4 to 7 letters and Class C are the words with a length between 1 and 3 letters. The Class is shown as a letter between parentheses, i.e. (A).

Values of zero percent are due to patterns not found in the corpus searched. If a pattern searched in the context corpus cannot be matched with a relevant result, the resulting precision is zero.

The first column to the left references the search algorithm used during a search for a word of different lengths or its phonetic interpretation. Results are posted according to equations (2) and (3). Notice that the Precision Performance of the phonetic search specifically, PDS is 100%. This indicates that when using a PDS the relevance of the retrieved data and the data retrieved is almost perfect. In other words, the search generated is very accurate because all the words pulled from the database searching phonetically are relevant in the Corpus database for the test word. If we consider the data for Class B words, it can be observed that PDS has a Precision of 100% again; however, the base line also retrieved all the relevant words; yet, the phonetic search did not have a single match. If we look further down at the Class C word, we noticed that PDS again has matched all the relevant data available, in this case for the Context Term data and the General Corpus.

Table 9: Phonetic Search: Strength & Weakness

	Word	Phoneme	PDS	DoubleMetaphone
Strength	Exact search	Exact search	Exact and approximate search	Exact and approximate search
	Fair Performance		Good Performance	Excellent Performance
			Better on Misspellings	Best on Misspellings
			Can find words ASR translated incorrectly	Can find words ASR translated incorrectly
			Performs Very Well with Class A & B	Performs Very Well with Class B
			Synthetic Voice (TTS) and ASR Independent	Synthetic Voice (TTS) and ASR Independent
	Language Independent	Language Independent	Language Independent	
		Numbers can be in words or numerals	Decodes numbers in any way	
Weakness	No Approximate Search	No Approximate Search	Approximate Search: Prone to false positives, higher as word get smaller.	Approximate Search: Prone to false positives, higher as word get outside the meta key code range.
		Underperforms with any class (A,B, C)		
	Sensitive to Syntax and misspellings	Sensitive to voice synthesis		
	Syntax errors produce no matches or mismatches	Voice Synthesis and ASR errors produce mismatches.		
	Performs fair with Class A,B & C	Underperforms with any class (A,B, C)	Sensitive to Class C	Sensitive to Class A & C
	ASR & TTS error variations cannot be matched	ASR% TTS error variations cannot be matched		
				Only English
	Numbers have to be identical in format	Numbers can be in words or numerals		Cannot decode Numbers

At a glance these results do not explain why the differences. The fact is that when the General Corpus was converted by the ASR, the original content suffered a transformation using the Windows 7 standard ASR. As the word "president" was converted to words and phonemes it lost its original since the ASR replaces randomly the phones "ih", "aa" and "ax". Sometimes caused by the intonation of the speaker, but in other instances it inserts "ax" to substitute any of the mentioned phones. Therefore, the word "President' can be found phonetically as one of the following morphemes:

(1) p r eh z ax d ax n t

(2) p r eh z ih d ax n t

(3) p r eh z ih d aa n t

Similarly when a Sythetic Voice is used to convert the sought word into a phoneme, again, the synthetic voice changes a few phones in the convertion. The first example (1) is the output of a Windows 7 voice sythesis “Anna”. The example below is the trasformation caused by a COTS voice from Cepstral ("Cepstral," 2010). Loading other voices and testing with different words form the bag of words revealed a pattern where “ax”, “aa”, and “ih” where being exchanged on for the same word. PDS in its algorithim replaces these characters for wildcards or “Don’t Care’s”. Heuristic observation demonstrates that in some intsances the as the word is converted to its phoneme, repeated characters are inserted to the ends of the word, another feature that PDS compensates. Then, we can now understand why the phoneme search for “p r eh z ax d ax n t” has a Precision value of 0% as well as a Recall value of 0%. It happens that the ASR conversion decided to encode “President” phonetically as “p r eh z ih d ax n t”. It is now obvious why the phonetic search could not find the phoneme. Likewise, the phonetic search Precision is 100% for the context term database, because this repository had encoded morphemes for “president” using a Speech Synthesizer voice that provided the morpheme “p r eh z ax d ax n t”. In this case, Precision was 100%.

In cases where there are unexpected phoneme substitutions, or misspellings, PDS will have greater Precision and Recall than the baseline. However for words of Class C, it does not use the wildcard, it does a phonetic search instead. This can be correlated with the search for the phonetic equivalent of PhD. Both, Phonetic and PDS have the same results.

The specific case of PhD is interesting since the word only appears once in the entire corpus as “Ph.d.s”. As we know, this is syntactically incorrect but semantically it defines a Doctor of Philosophy. The ASR translation decided to convert to the odd spelling of a known acronym. Certainly, the baseline word search cannot find the acronym unless you typed exactly the same, “Ph.d.s”. Nonetheless, PDS and the phonetic search will find it because the phonetic equivalent is “p iy ey ch d iy” regardless of how you type it, i.e. “PhD”, “P.H.D”, or “Ph.d.s”. Notice how the powerful Double Metaphone failed to retrieve a single match. In fact, Double Metaphone retrieved 93 words, none of them a relevant match. This is an example of critical information that only PDS can retrieve. The other three search methods exposed in this document are incapable of finding such word caused by a transformation of the original content by the ASR. The Metaphone search on this same task after searching for “ph.d.s”, “Ph h ds” and “phd” retrieved 7, 13 and 98 matches for key codes FTS, PTS & FT respectively, but none were relevant, a were false positives. The ability to of an ASR to change a word beyond user recognition while keeping its phonetic equivalent, makes it very difficult to find relevant matches while using any type of query, such Database queries need to be modified in real-time for acceptable performance; PDS offers a solution to the nuances of ASR translation.

On Table 9, we expose the weaknesses and strengths of each search algorithm. Each search algorithm has its own advantages. Particularly PDS is versatile; it is language independent; because it searches through phonemes, the algorithm can be applied to any phonetic set. It also performs well with misspelling and provides the flexibility of an approximate search where it will recall relevant words to the original search without falling into large false positive recalls. It performs well in Class A & B. For Class C, PDS also performs an unmodified phonetic search with better results in this Class than PDS. PDS is the only algorithm that will work well with numbers in both number format, and letter format combined. Because it converts the words to phonemes with the use of a TTS, it is a modified phonetic search after that with good results.

The Double Metaphone Algorithm is very powerful In a Class B environment. It suffers from retrieving false positives, which affects its Precision because not all the words retrieves may be relevant. PDS is not affected by the false positives unless in Class C, where small words repeat themselves in unrelated longer words causing mismatches. The specific case of the "PhD" word is a unique example, but word such as "See" and "Sea" will generate the same Morpheme, and "watchdog" and "dog" will most likely be matches for "d aa g" or "d ax g". I the future will like to add to the wildcard PDS, avoidance to all the small articles and common Nouns such as "The" or "she" and therefore increase the performance of PDS.

Another interesting comparison test evaluated is the percent of improvement over the base line test. Factual data can be studied in appendix B; however, on the following table

Table 10: Baseline and Phonetic Search Comparison

Test Word	Baseline % Improvement Context + Corpus	Word Size Class	Query Heuristics	Susceptibility to Complete Irrelevant Hits
Word	0.00%	A	NA	No
Phoneme	-98.85%	A	NA	No
PDS	2.31%	A	Phoneme	No, except for Class C
Meta	12.68%	A	Letters	Yes
Word	0.00%	B	NA	No
Phoneme	50.00%	B	NA	No
PDS	50.00%	B	Phoneme	No, except for Class C
Meta	250.00%	B	Letters	Yes
Word	0.00%	C	NA	No
Phoneme	Undefined	C	NA	No
PDS	Undefined	C	Phoneme	No, except for Class C
Meta	Undefined	C	Letters	Yes

the reader can observe the large improvements over the baseline. On Class A word size the PDS shows improvements of 2.31% and 50% for Class B. The best results show an increase of 366.67% over the baseline in comparisons where the word search relevant hits vs. PDS was 6/28. The Baseline Percent Improvement is calculated in the following equation (9).

$$Baseline\ \%\ Improvement = \frac{Test\ Search\ Relevant\ Hits - Word\ Relevant\ Hits}{Word\ Relevant\ Hits} \quad (9)$$

Best results using the Metaphone for relevant results did not exceed 250% improvement; nevertheless many times better than a word search when it comes to retrieving relevant results.

Values shown as “Undefined” are cases where the word search did not find any results, therefore the operation result is infinity; none of the evaluated searches is infinitely better than the other, yet were able to find relevant results that the baseline word search could not. We also need to remind the reader that the Metaphone search has been adapted to search through words and has not been used for this purpose, it design was intended to find names in records that may present errors due to spelling or human keying errors.

Briefly, the results of word spotting using phonetic searches prove to be better than the baseline in most cases. The optimal solution would analyze the word sought before the search and define the best approach to be used based on the structure of the word and phoneme. The selected approach could be the outcome of many searches in parallel as the results are analyzed for relevancy. Only the best results will be used to present the best solution to the user. The reasoning behind this hypothetical solution is based on the results of the test done on word spotting with different phonetic searches. It was observed that the benefits of one model do not cover all the spectrum of word lengths defined as Class A, Class B and Class C. PDS search is better when relevant results are needed and the word is bigger than five characters. Similarly, Metaphone search works better for Class B; however, this could be improved by varying the length of the Key code extraction. The setback of this approach is that all the corpus will have to be server indexed several times as a new length is introduced in the Metaphone key extraction algorithm. At the search client side, the word length shall determine the length of the key code used for the Metaphone search key code extraction and provide different queries that will explore the different key code sets. The phonetic search alone without modification has demonstrated unreliable. It has shown 50% improvements over the baseline in very few occasions. Most of the results are negative as compared to the baseline; therefore not reliable because of the changes in the phonetic streams caused by the ASR. Experiments are being done where the corpus is check for error before search based on mouth movement using MPEG4 (Aleksic & Katsaggelos, 2004); an interesting approach but slow for any large volume application.

Metaphone phonetic search is very fast and reliable in most cases; exploration with the key code generation size can improve the unrelated retrieved information at the cost of larger repository space. A new addition to the Metaphone algorithm is the ability to trace back to the original content and the location of each match, indeed a necessary feature if the video is to be searched based on the word search.

PDS is a step forward in the creation of preemptive search that addresses the errors caused by speech recognition and syntheses while it scores well retrieving relevant results. It is language independent and works well for most word lengths. Applications that require sorting through large libraries of video with a method to search the video by words could benefit from the PDS.

The solutions offered all require a pre-indexing of the data, is here where the conversion of all the videos is done. To do this efficiently parallelizing ASR should be explored to speed up the conversion operation.

CHAPTER EIGHT: CONCLUSION

We consider most advantageous, the capability of our system to provide summarized ASR word content to map where the topics of interest reside within the video. This is a vital feature when skimming trough large video archives. Indeed, proposed system can provide meaningful summaries in addition to isolated words, as we diminish search latency because each queries are done on video summarized data rather than the complete video ASR data which has been demonstrated to be more efficient. Furthermore, it is possible to have a two-level search where initially the context of the video is searched efficiently and presented to the client summarized using web-based technologies. On a second pass, after the user selects a video from the presented list of corresponding results, the client can further assist the selection by searching through the previously indexed ASR translated data aligned with the video.

The Indexing Schema can provide the necessary data for video correlation. The positive relevant hits returned from every search, contain index information that potentially allow a positional alignment of the results of the search with the original video in time, conceivably avoiding random audio playback exploration of the entire video. Instead, video snippets can be played on-demand for each match, optimizing the work needed to find specific information within the video. This allows a new more effective the user interface experience altogether.

In the case of searching through dirty data, lost words to ASR Translation are not of great significance if they repeat themselves in the entire content of the ASR transcript. Due to inherent ASR translation errors and noisy corrupted material, some of the original video transcript content may have been lost or transformed, but in the newscast material used, enough information is always available to discern at a glance if the selected video is of the user's interest. Perhaps, most useful is the ability to traverse through large video content with ease regardless of WER. Notice that in informational indexing, WER becomes less significant, since we focus in providing referential mapping of the searched content rather than an exact mapping between the ASR translation and the video. The goal is to be able to distinguish sections in the video of user interest. The presented approach provides the infrastructure to make this possible.

In that case, it is possible to augment the video description by using the context terms list of words φ used to describe the video through summarization, by inserting different OOV words, but that carry a similar semantic connotation.

PDS is not too sensitive to changes in syntax; however, the sound of the word is key for the effectiveness of any phonetic search. It is possible that in some cases, misspellings on searched words can be also found with PDS, since phonemes preserve the sound of the translated word. Therefore, increasing the OOV words as part of the video index metadata during the preprocessing of the video data, further increases the possibility of positive matches. The reader shall note that the translated video content and its word positional and frequency data obtained using the Aho-Corasic algorithm, is kept intact and available to align the textual information with the video content.

Indeed, the devil is in the details. The time alignment of the data with the video considering all the deletions and substitutions caused by the ASR translation, conversational inaccuracies and background noise, is a mere approximation without original manual translations of the video source. The presented indexation allows mapping the phonetic content back to the originating word by location. Its alignment with time can be done approximately enough to provide client user video skimming. Nevertheless, if our goal is to categorize video, optimize, and enhance the search though phonetic transcription data, we do not need a captious remark to realize the alignment between the word/phoneme position and the video timeline will be a few seconds skewed.

Another interesting fact that evolved from this research is that the PDS is able to find video content that was lost during translation. Perhaps, the translation can sometimes retain phonetic variations of certain words that can only be perceived through phonemes.

Another remarkable result is the ability to leverage the fact that different speech recognizers and synthesizers do not standardize their phonetic symbols. However, variations of phonetic transcriptions are indirectly induced into the search data because within the same manufacturer, the phonemes generated by the speech recognizer do not necessarily match the phonemes generated by that same vendor's synthesizer. This can create a challenge to deal with a phoneme mismatch during exploration. This disparity becomes a pattern when different voices are used concurrently at the synthesizer to convert difficult words. The common denominator is mismatched phonemes can be replaced by wildcard parameters used to search the corpus with positive results as shown herein.

We also noticed that OOV additions due to translation errors result in grammatically incorrect words and phonemes. In some particular cases, letters were repeated sequentially two or more times within a word or phoneme due to ASR or synthesizer stutter. We had to correct for these errors during the search synthesis process to obtain positive results.

The search interface used to demonstrate the PDS concept is not an optimized search engine. We realize that word spotting does not consider many words. A Multi-word search can be done in many ways using different factors, but finding which method is the optimal depends by large on the interest of the client. The implementation of an efficient search engine is a current topic of research and beyond the scope of this report; therefore we limited the results of word spotting to demonstrate disparity between Phonetic, PDS, and Metaphone concepts compared with the baseline word search.

Nevertheless, we did experiment with two and three words and found that different multiple word searches can generate unusual results. In any case the phonetic search is capable of finding words contained one after each the other, but incapable of finding all the related words contained in the search without repetitive searches.

The paradigm opens endless possibilities by adding Levenshtein distance calculations, HMM and Viterbi algorithms to estimate which nearby words or patterns are relevant; again are we interested in searching words in a close distance to each other, or words that are relevant within the original text; the answer is both. A shortcut solution can be constructed using logical AND to concatenate the phonetic transcriptions of multiple words, but the search returns false positives a significant portion of the time. Multiple word searches go beyond word spotting by increasing the complexity of the search due to the grammatical content of a search sentence. The paradox arises when a multiple word search is used to compare results of a search, because the relevance of the results is not just tied to the word meaning but to the meaning of the sentence perhaps.

The inclusion of slider bars that enhanced the ability to increase and decrease the phonemes relevant for a search allowed us to determine the errors found die to speech recognition. This feature mixed with the use of simultaneous Speech Synthesis voices gave us the insight necessary to discover the distortion patterns that PDS handles.

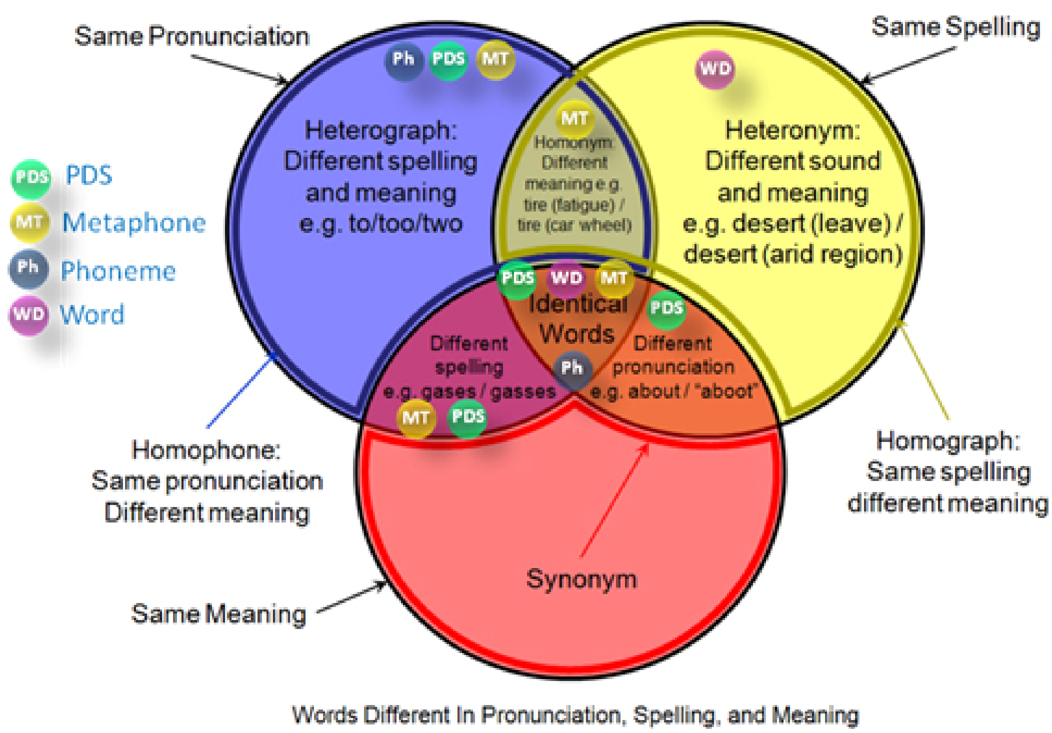

Figure 24: Phonetic and Word Search Space Comparison Homonym ("Homonym," 2010))

The search space comparison figure (Figure 24), depicts the sets of words where the PDS, Metaphones, Phonetic and Word searches are most likely to generate relevant results. It can be observed that none of the searches studied can retrieve related words form the read search space, words that have the same meaning but are pronounced and spelled differently. This case requires a grammatical analysis of the words and a Synonym dictionary to verify the relevance of a sub set of words. As this set is defined, then each word can be searched using Aho-Corasic for relevance; however, this is not a phonetic search and not considered in this work. Nevertheless, all the other areas of Homonyms can be coved with a hybrid search that includes the phonetic search methods studied.

If we consider the PDS search using the categorization in Figure 24, we can add that PDS can search for misspelled words, Homophones, Homonyms, Heteronym and identical words. Similarly, Metaphone search can find Homophones, Homonyms, Heteronyms, and identical words, in some cases, at the cost of irrelevant words (False Positives). Searches that are difficult to perform using a word search without phonetic or grammatical analysis of the word sought. Matching identical words, any search can perform, but as the sound of the word is factored into the search, phonetic analysis will perform better because it preserves the sound. As we add into the mix ASR conversion errors, PDS has the capability of finding word with some misspelling or miss sounding that are relevant, but the Precision will diminish. Metaphone search is this case will search the words within a key code and most likely return a large set of unrelated data. On words with the same pronunciation and different meaning both Metaphone and PDS will achieve good results and bring into the set other related words. Phonetic search will only do an identical match as well as the baseline search, avoiding all the data that was changed during the ASR conversion.

The beginning to end architecture proposed for indexing and searching video content can be scaled to larger systems and support large libraries of video given the opportunity of a large-scale system. The database uses is an enterprise version of SQL server that supports farming and replication. Multiple coincident searches form multiple distant clients should not present significant delays. However, the indexing and phonetic retrieval of information can be done much faster with the use of LVCSR engines that can process multiple audio streams at a time, enhancing the system's ability to process any video steam automatically.

The Multi-search interface serves as an experimental client to the database to evaluate the effectiveness of the algorithms proposed. It provides the ability to perform word and phonetic queries onto three different databases while modifying each search progressively to evaluate the results. All the results stored back into the DB with analytical calculations added. Moreover, after each search is performed, the data is presented locally for research analytics. By no means shall this Multi-search interface be considered the front end of a client interface such as Google search or any other search engine, however presents an addition to such interfaces since it allows categorization of video, a topic in vogue but with not a lot of enterprise solutions.

Future work

Is of the interest of this researcher to add the experimental lessons learned from this research to an enterprise level experiment. It would be interesting to receive the funding necessary to implement a small-scaled system that is web-based and supported by a server farm that will perform video striping and indexing around the clock. As the content is organized in multiple databases, two client interfaces exist. A video posting interface that allows posting video into the system and another web-based interface that uses a search engine API with the enhancements suggested to search for related video, based on the phonetic interpretation of the videos stored.

We know that the videos stored need to have voice in them and that many will be impossible to decode, but the ASR shall pull some words that are correct if it contains words, most do, the ones that are silent can be avoided for the experiment. Then after the website is posted, let the users use the system worldwide to discover the real inefficiencies. Following spiral software engineering cycle, improve the system every month. As the system matures, move to be used to by existing video libraries, such as the recorded courses in an educational institution and explore usability of the system.

Then, it can be mass-produced to address military and business intelligence decoding calls from call centers of phone taps within others. It can be used to categorize a library of historical speeches or broadcast material regardless of the language since the changes required are small if an existing ASR for the studied language is available.

The ideal search algorithm may have to include features from not only the phonetic interpretation (PDS or Metaphone) or textual data (Word), bit also grammatical relevance, not considered in this study. The experimental results demonstrate that a hybrid algorithm that considers different phonetic and syntactic methods will work best, and can be varied with the needs of the client automatically.

The Metaphone search can be explored further by varying the key code length with the length of the word automatically, therefore generating key codes that represent longer words. The drawback of this approach is that the corpus will have to be indexed periodically to accommodate new word lengths; a different set of key codes requires corpus duplication for every word contained, expensive in fact. Our small 5-hour video sample generated over 97500 records needed for Metaphone operation. Exploring with variable lengths of five different Class word lengths will require about 390000 records to maintain.

PDS search did not fare well with Class C word lengths. During indexation the words the words that do not add descriptive value to a topic such as “at”, “the”, “it” or conjunctions and prepositions within others, can be avoided as part of the search space because not only they are difficult to find, but also delay the result. False positives can be diminished by this approach.

Exploration with non-synthetic voice phonetic convertors can increase the reliability of the search caused by the STT conversions. Currently we convert every word used to its phonetic equivalent using SAPI compatible voices, but inaccuracies have been discovered using same vendor voices that add error the corpus. Using a SAPI compatible silent word to phone converter could speed the process and provide better reliability.

On the ASR side of the conversion, the system can benefit from the use of professional Speech Recognition Engines that allow multi-threaded operations and massive training. Such devices have demonstrated better WER and speedup the indexing operation.

On august 10, 2010 a patent was filed The Georgia Tech Research Corporation for phonetic searching using phonemes. The approach is much simpler than the proposed architecture (Cardillo, Clements, & Miller, 2010). We believe that a submission for patent application is possible for this approach and worth considering given difference in expression with current phonetic approaches.

APPENDIX A: VIDEO WER RESULTS

WE ARE back and joined now by henry paulson the former treasury secretary and alan greenspan former chairman of the FEDERAL reserve WELCOME both OF YOU back TO MEET the PRESS

*** WE'RE back and joined now by henry paulson the former treasury secretary and alan greenspan former chairman of the *** reserve OPEN both *** THE back TWO IN the DEPRESSED

AH dr greenspan HERE WAS THE headline in THE NEW york times yesterday ON that friday jobs report

OF dr greenspan *** HAS A headline in *** *** york times yesterday *** that friday jobs report

A it was this JOBS STRAIGHT FALL TO 9 7 PERCENT

AND it was this *** JOBLESS RATE FALTA 9 *** 7%

giving hope that the worst is over

giving hope that the worst is over

IS THIS jobs report *** *** SIGNAL a TURNAROUND?

THIS IS jobs report SIGNALED THAT TURNER a GOOD

IT DOESN'T SAY IT WILL TURNAROUND BUT WHAT IT DOES SAY IS THAT A the TURNAROUND WHICH HAS ALREADY OCCURRED

*** *** *** *** PERSON FOR TWO MORE AND MORE WOMEN ARE AFRAID OF the *** INTERNAL USE ONLY FOR

is moving

is moving

BUT NOT IN ANY AGGRESSIVE MANNER

*** *** *** TO A LOAN

and *** *** *** *** *** and THE

and THE ISSUE YOU A HAND and A

secretary *** PAULSON if you look at the jobs lost since the recession began

secretary PAUL SINGH if you look at the jobs lost since the recession began

8 4 AH million jobs over that time horizon

8 *** 45 million jobs over that time horizon

AH the question is UH what IS GOING TO cause A TURNAROUND WHEN DO you see this UH THIS jobless rate

OF the question is WILL what *** SCAN IT cause THE TURN AROUND AND you see this SET IS jobless rate

actually stay in the single digits

actually stay in the single digits

well *** the economy is *** clearly recovering

well TO the economy is THIS clearly recovering

and i have UH great confidence *** THAT we have such a dynamic private sector ON THIS UH in this country *** THAT THEIR EVENTUALLY GOING to begin

and i have A great confidence IN WHAT we have such a dynamic private sector AN ISSUE THAT in this country THE ERROR OF THE JUANCA to begin

creating jobs

creating jobs

now one of the factors not the only factor but one IF the factors that will help

now one of the factors not the only factor but one OF the factors that will help

*** IF more certainty

IT HAS more certainty

AH   with regard to AH TO actions OUT of washington and for instance

THAT with regard to THE TWO actions *** of washington and for instance

AH CERTINTY   with regard to AH financial regulatory reform

THE UNCERTAINTY with regard to THE financial regulatory reform

WELL UH   UH   will help

LAW THAT ALONE will help

A   AND AND in   terms of   not just ***   REGULATORY REFORM WHAT WE TALKED about DR GREENSPAN BUT also just the idea *** of

*** IN IN in terms of not just RETURNED FROM   A   WEB OF TALK   about THE KOREANS   HAS also just the idea OF of

A   the notion of what the government can do now FOR   REGARD JOB SPILL AND other things to bring down unemployment more STEADILY

*** the notion of what the government can do now WE'LL SURVIVE WITH JOBS STILL other things to bring down unemployment more STEP

I THINK we have to start WITH A focus

IS THAT we have to start YOUR FREE focus

of *** economic activity IN OTHER WORDS

of ITS economic activity *** *** COVERAGE

jobs are created by having *** to do YOU SEE you CAN'T put jobs

jobs are created by having SOMETHING to do *** IF you CAN put jobs

BEFORE economic activity

FOR economic activity

and *** *** I WILL THEREFORE ARGUE WHAT WILL BE most useful AT THIS PARTICULAR stage

and TYRE WITH THEIR FULL OF YOUR LOVE WITH THE most useful IS THE TRUE stage

is cutting taxes on small business BECAUSE THEY ARE THE BIG creator of jobs is cutting taxes on small business THAT WAS A OVER THE creator of jobs

but they won't hire anybody IF THEY DON'T have ANY BUSINESS

but they won't hire anybody *** *** TO have *** ***

SO YOU HAVE to GET THEN to act in a manner

*** SOME HALF to GIVE THEM to act in a manner

which creates THE types OF economic activity

which creates AND types AND economic activity

which DRAW IN AN ever increasing demand for LABOR

which *** TRULY AND ever increasing demand for IT

and that is a question in terms of WHAT IS happening out there where where is the IMPUTES

and that is a question in terms of *** WHAT'S happening out there where where is the IMPETUS

for businesses to start hiring AGAIN

for businesses to start hiring ***

*** *** WELL AGAIN

OF A GUY AND

I i just believe SO MUCH IN HOW dynamic our economic system is AND our economy IS

*** i just believe *** SELMER SCAN OF dynamic our economic system is IN our economy ***

*** the one thing i know for sure

IS the one thing i know for sure

is it with the economy

is it with the economy

*** recovering

IS recovering

ultimately

ultimately

the private sector will do what needs to be DONE

the private sector will do what needs to be ***

and create opportunities and jobs

and create opportunities and jobs

I I agree with alan that the *** UH

*** TO agree with alan that the UP THAT

THAT when you look at a job TO BUILD THERE ARE SORTS OF THINGS THAT THE CONGRESS should be focusing on

*** when you look at a job *** *** *** SHOULD ALTER THAT THE SEARCHER FINDS IT, should be focusing on

ARE temporary incentives *** ***

OUR temporary incentives FOR BUSINESS

FOR BUSINESS TO UH TO ATTIRE

*** *** TWO LET THE DOLLAR

A yet is that enough if THERE IS not a business willing to take the risk to EXTEND EXPAND

AND yet is that enough if *** THERE'S not a business willing to take the risk to ITS TACTICS

*** *** AGAIN AS I SAID EARLIER

THAT WE'RE ALL THINKING AND THAT IS

*** *** *** part of *** IT IS

ON SEVERAL ARE part of THAT HE HAS

confidence and psychology what's going on

confidence and psychology what's going on

IN inside THE the HEAD OF the ceo

AND inside THAT the *** TANNER the ceo

AH IN HOW COMFORTABLE DOES HE

*** *** OF AN UNCOMFORTABLE ACHIEVE

AT HE OR SEE she feel about *** *** *** the FUTURE

THAT THE RECEIPT IS she feel about THE UP THAT the FEATURE

but *** *** AGAIN

but THAT HE AND

it's very difficult to sit here

it's very difficult to sit here

and *** SAY NOW where *** IS the economic ACTIVITY GONNA come from which *** AREA which *** BUSINESS but IT always does COME

and SO THEY KNOW where HE HAS the economic *** ACTIVITY, come from which CARRY OUT which HAS LIST but HE always does ***

and it WILL COME

and it *** WILL,

A WILL have stable *** *** financial markets and a recovering economy it's GOING TO take some time THOUGH

THAT WE have stable FOR THE financial markets and a recovering economy it's *** GONNA take some time TO

WHEN is the recession IS over

SUBMIT is the recession *** over

*** recession is over IT

THE recession is over ***

*** BOTTOMED THE back in the middle of last year

THE BOTTOM ARE back in the middle of last year

AN WHILE IT DOESN'T have the strong momentum I'D hope IT WILL have

AND LOWER CASH AND have the strong momentum HONORED hope YOU WOULD have

STRANGELY BECAUSE OF the fact that we HAVE A STRONG FORTH quarter

*** SPURNED FOR the fact that we IN CITRUS GROWN FOURTH quarter

which was essentially using up A LOT of the *** MAKING power of *** EVENTS

which was essentially using up FROM ONE of the LATE IN power of THE GIANTS

WHICH WAS A gradual reduction in the RATE of THE DECLINING inventories

*** PITCHERS FOR gradual reduction in the WAKE of DECLINE IN inventories

we DID IT ALL IN the fourth quarter

we *** *** *** DID: the fourth quarter

AND WE SENSIBLY we SHOT OUR AMMUNITION

FOR RESEARCH AND we SHOT, AND IMAGE

*** *** *** SO IT'S GOING TO BE SLOW TREADING THING

FOR REFORM OF THIS SHALL OLD CHURCH AND TRADE ARE DUE

BUT I DO THINK WE WILL BE moving forward

*** *** FOR A WEEK OF THE moving forward

and AS HANK SAYS THE ISSUE HERE is *** *** BASICALY innovation

and THE TIME SHARES AND ISSUED YOUR is BASICALLY AN OVERAGE innovation

INNOVATION by DEFINITION

*** by DESTINATION

is not *** FORECASTABLE

is not FOR FESTIVAL

SO we don't know where the JOBS ARE COMING FROM

SURE we don't know where the *** *** CHILD CARE

WE DON'T KNOW HOW this market is EXACTLY IN TERMS OF

AND THE BUILDING OF this market is IN FACT WE INTERNSHIP

dynamics GONE NU GO move forward

dynamics CAN DO TO move forward

but we know that THIS PROCESS is UNDER WAY

but we know that THE PROSECUTION is *** UNDERWAY

and THERE IS EVERY REASON to believe IT will CONTINUED

and *** *** *** CONDITIONED to believe *** will CONTINUE

*** AND you look at the stock market THAT THE FACT that ITS BEEN on a downward path FOR the past couple OF weeks

TO MAKE you look at the stock market *** *** *** that SECOND SPAN on a downward path OF the past couple *** weeks

A DOWN over SIX PERCENT A since january

OF DOWNED over *** 6% BUT since january

WHAT KIND OF warning sign IS is that

*** *** OH, warning sign THIS is that

WELL IS more than A WARNING SIGN a ITS

*** WAS more than 1 INCH AND a FUTURE

important to remember

important to remember

that

that

*** *** EQUITY VALUES stock prices are not just paper profits

SET THE DRIVE THE stock prices are not just paper profits

they actually have a profoundly important *** *** IMPACT IN economic activity

they actually have a profoundly important THAN THAT FOR AN economic activity

and if stock prices start continuing down

and if stock prices start continuing down

I WOULD GET VERY CONCERNED

*** A BIT FOR INJURED

i agree with that but i also never placed TO much emphasis

i agree with that but i also never placed TOO much emphasis

AND what the market DOES for any WEEK OR TWO

ON what the market GOES for any *** WEAKER TO

YOU NEED to really look at THIS JUST like YOU look at ACADEMIC DATA over a period OF TIME

THE CITY to really look at THE ASIS like TO look at THAT, THAT over a period *** CLIMBED

AND IF YOU look at this over a REASONABLE PERIOD OF TIME

*** *** 30 look at this over a REASONABLE, JUST FIVE WEEKS

WE WE seen a UH

*** WE'VE seen a ***

A A A very solid ***

*** *** *** very solid THE

stock market

stock market

LET ME ask you A about

*** LILLY ask you ARE about

the president AND about the president's team this IS SOMETHING you wrote in YOU new book on the brink

the president *** about the president's team this *** SUMMER you wrote in YOUR new book on the brink

A about election night

OF about election night

AND a change OF leadership after the democratic candidate was declared the winner AT ELEVEN pm YOU WROTE WENDY YOUR WIFE

IN a change IN leadership after the democratic candidate was declared the winner OF 11 pm THE ROAD WHEN THE LIGHTS

WOKE ME up to tell me the historic news i went back to sleep CONFIDENT by the knowledge that OUR president *** ELECT

WILL BE up to tell me the historic news i went back to sleep COMFORTED by the knowledge that A president WHO LACKED

was *** BARAK obama fully understood the threat our economy still FACED

was BROUGHT THE obama fully understood the threat our economy still FIXED

what DO you say now after more than a year is THAT confidence STILL HIGH IN HIM AND ITS TEAM

what *** you say now after more than a year is A confidence *** *** *** *** STILL-IT ISN'T ALL

*** *** *** BUT I WUWATA WATA I SAY IS THIS

WANT A LOT WORSE IN AS AS THE ZIPPED OF THE

THE i TAKE A real comfort IN the fact that

*** i THINK THE real comfort AND the fact that

the programs THEY were put in place to *** STABILIZE THE ECONOMY

the programs THAT were put in place to SETTLE A CIA THE

WERE continued and AH much of WHAT'S BEEN DONE WAS A continuation or *** LOGIC extension of those programs

WORK continued and OF much of *** A WHITE MAN, THE continuation or A LOGICAL extension of those programs

I believe *** the *** UM financial markets are stable

TO believe THAT the OF THE financial markets are stable

I believe the programs have *** WORKED THEY PREVENTED the COLLAPSE of the *** *** the financial markets *** *** A PREVENTED a real catastrophe I THINK WE COULD'VE HAD TWENTY FIVE PERCENT U UNEMPLOYMENT IF EIF IF IF IF the system HAD collapsed

OF believe the programs have WORK TO DATE THAT the CONTENT of the COLLAPSED OF the financial markets THAT THE CONTENT IN a real catastrophe *** *** OF THE WEEK OF THE 25% UPON EMPLOYMENT DATA FOR THE DEFENSE AS the system THAT collapsed

and i believe that WE ARE GOING TO see that every penny that's been put in the banks

and i believe that *** *** WE'RE GONNA see that every penny that's been put in the banks

IS GONNA come back

THAT CAN come back

with *** INTEREST so i think THAT MONEY IS coming back *** SO AT and that was WHAT i *** *** was talking about

with A TRIP so i think *** THE MONEY'S coming back SOME OF THAT and that was WHEN i WAS WHAT was talking about

ON ELECTION EVE because

LIKE TO LEAVE because

both presidential candidates

both presidential candidates

had supported

had supported

the TARP legislation and i think that was critical IF THEY HADN?T we would've been defenseless

the ENTIRE legislation and i think that was critical OF THING AND we would've been defenseless

BUT YOU CERTAINLY seem depressed reading your book with candidate obama FRANKLY more so than senator mccain DID YOU VOTE for obama U WELL IICUCK

BY THE SURVEY seem depressed reading your book with candidate obama FRIDAY more so than senator mccain *** *** DIGITAL for obama *** *** ***

*** *** *** *** WHO WHO I VOTED FOR

WILL HAVE THE WILL TO LIVE WITH THAT FORBIDS

BUT IS between me and the VOTING BOOTH BUT THE

*** *** between me and the *** *** CALLING OF

BUT the A

*** the THE

*** *** *** *** but *** there's

THE THE UP TO but IF there's

I   was very impressed *** ***   THAT CANDIDATE obama

TIME was very impressed THE BREAK AND THE   obama

was

was

very concerned WITH what was going on

very concerned *** what was going on

and *** *** WAS was very supportive

and TO THE WAYS was very supportive

*** A   candidate *** MCCAIN

OF THE candidate WE CAME

I WILL ADMIT GAVE ME A FEW MORE anxious days and hours

*** *** OF A LIMIT THE DFG WERE anxious days and hours

BUT I WILL ALSO SAY THAT AS HE was falling behind in the polls

*** *** *** OF COLOSSAL STRAIGHT AND SHE was falling behind in the polls

*** I WOULD'VE BEEN very easy for him TO DEMAGOGUE

TO ONE OF THEM very easy for him TO, GOT

that issue PLAYED TO the POPULOUS card

that issue TO PLAY the POPULACE card

and if HE'D come out against

and if YOU come out against

what WE WERE trying to do

what *** WE'RE trying to do

we WOULD'VE GOT IT I BELIEVE WE WOULD'VE HAD the TARP legislation passed and we would've BEEN LEFT DEFENSELESS we *** *** WOULD HAVE GOTTEN OUT LATELY WITH the AVATAR legislation passed and we would've DONE LAST DEFENSE

*** *** DR GREENSPAN one more question about jobs do you think that unemployment rate goes up again before it comes down

WAS THAT A RECENT one more question about jobs do you think that unemployment rate goes up again before it comes down

I AM NOT SURE NOR ONE OF THE REASONS IS the official data OF unemployment

*** *** *** *** FROM ENSURE AN ORDERLY SHEET AND the official data ON unemployment

is a sample

is a sample

and IT FLUCTUATES AS WE OBSERVE THEN IN THE JANUARY report

and THAT FLUCTUATE TO ASSUME WE OBSERVED TO AN INPCAMERA report

*** LITTLE literally took its

YOU WILL literally took its

*** *** SERIOUSLY IS THAT the EXACT numbers

TWO ARE USUALLY RESTS WITH the TRACKED numbers

THERE WAS SEVEN HUNDRED and EIGHTY FOUR THOUSAND JOB INCREASE IN JANUARY NOW that DIDN?T HAPPEN

*** *** *** TWO and *** 784,000 JOBS AND CREATES AGENDA THAT FACT that *** ENSURE

A SO that WHAT we can expect is A BACKING IN FILLER I THINK WE ARE GOING TO STAY approximately

*** *** that WILL we can expect is *** AN ACTION FOR OUR HOUSE IN ORDER TO STATE THAT approximately

the nine to ten percent LEVEL HERE

the nine to ten percent OF THEIR

FOR GOOD and PROBABLY THE REST of THIS YEAR

*** PROBE and *** *** PROGRESS of THE FEUD with the sole exception of *** THAT PERIOD WHEN THEY start to hire a VERY large number of *** CENSUS WORKERS

with the sole exception of THE KOREAN WAR AND THE start to hire a FAIRLY large number of CENTERS FOR IMAGE

REMEMBER this IS THE DECENNIAL CENSUS and THAT IS GOING TO HAVE SOME POSITIVE EFFECT BUT IS VERY DIFFICULT TO MAKE THE CASE THAT UNEMPLOYMENT IS coming down *** ANYTIME SOON

OF this *** *** *** *** and THIS AND THE SENSES AND THAT'S COMMISSION PRESIDENT FOR FOUR AT THAT THOROUGHLY DIFFERENT METHODS THAT CAN CLONE THIS coming down AND TIME TO

let me ask you about housing A DISTURBING report on wednesday in the new york times TALKS about people *** *** UNDERWATER IN their *** MORTGAGES the number of americans

let me ask you about housing AND STARVING report on wednesday in the new york times TALK about people ON THE WATER AND their MORTGAGE IS the number of americans

the paper reported WHO OWNED more than THEIR home WAS WORTH WAS

the paper reported *** LOAD more than A home *** TO WORK

*** virtually NIL WHEN the *** REAL STATE COLLAPSED in *** 2006

WAS virtually NO LIMIT the LIST A CLASS BEGAN in MID 2006

ABOUT the third quarter *** 09 AN estimated 4 5 million HOME OWNERS HAD reached

ABOVE the third quarter OF NINE AND estimated 4 5 million *** HOMEOWNERS WHO reached

the critical threshold with THEIR HOME VALUES DROPPED MORE THAN 75 PERCENT

the critical threshold with A RANK THE HOME'S VALUE DROPPING BELOW 75%

of the mortgage BOUNDS

of the mortgage BALANCE

WERE are now AT the point of maximum vulnerability

WE are now *** the point of maximum vulnerability

THAT'S ACCORDING TO SAM KADER a senior economist with first american *** CORELOGIC the firm that conducted the *** research

*** THAT SPLIT ITS ENCOUNTER a senior economist with first american CORE LOGIC the firm that conducted the RECENT research

people's emotional attachment to their property is melting into the air *** *** SECRETARY PAULSON

people's emotional attachment to their property is melting into the air SEARCH A PULSE OF

what happens if housing prices go down *** AGAIN

what happens if housing prices go down THE DEBT

when YOU already got THIS KIND OF PRECARIOUS situation

when YOU'VE already got *** THIS, TO CARRY situation

IT clearly WOULDN'T BE GOOD I AM not predicting that BUT WHAT i I THINK this issue *** IS is A A CRITICAL important one

IS clearly *** WHEN THE KIND OF not predicting that BY BLOOD, i THINK THAT this issue HAS TO is THAT THE CRITICALLY important one because ITS very difficult for governments to design a program that IS GOING TO BE effective and GOING TO BE fair to TAX PAYERS

because IT'S very difficult for governments to design a program that *** HAS GONE INTO effective and *** *** *** fair to *** TAXPAYERS

AH a program to keep people in their homes

OF a program to keep people in their homes

if THEY DON'T WANT TO STAY in THEIR HOMES

if *** *** NO ONE STATE in THE HOPES

*** AND SO A BIG part of WHAT WE focused on was that BRINGING the private sector together to keep THOSE INTO THEIR homes that can afford to stay IN THEIR HOMES AND WANTED to stay there

TO SELL IT TO BE part of AN LIKELY focused on was that BURY the private sector together to keep THE ROSE AND homes that can afford to stay *** AT HOME TO WANT to stay there

now

now

*** WHEN YOU LOOK at

HE LAN WHO WORK at

the CRISIS

the CONCEPTS

I THINK that PART OF the reason

*** *** that *** ARE the reason

that so many experts *** so many people didn't FORESEE housing

that so many experts THAT so many people didn't RECEIVE housing

AS BEING the cause AND AND AND AND COUNT ME among those was that if you look at

HAS BEEN the cause *** *** AN END TO THAT, among those was that if you look at

our country since *** WORLD WAR two

our country since IT WON'T WORK two

residential housing prices

residential housing prices

*** have generally gone up

THEY have generally gone up

*** we haven't HAVE nationwide *** DECLINES and MORTGAGES HAVE BEEN GENERALLY perceived to be safe investments

THE we haven't HAD nationwide TO CLIENTS and THE MORTGAGES INTERNALLY AND perceived to be safe investments

*** SO when WE get the kind of *** DECLINE WE had in housing prices

SINCE WER when YOU get the kind of THE CALLING WE'VE had in housing prices

that *** *** really *** *** really destroys wealth across the country but *** also changes behavior

that IT CAN really THINK THAT really destroys wealth across the country but IT also changes behavior

because historically

because historically

*** EVERYONE WHO HAD a mortgage would

TO EVERY ONE THAT a mortgage would

CRAWL FIGHT DO WHATEVER IT TOOK

*** *** *** QUALIFY DELIVER TALK

to make the mortgage payment AND avoid

to make the mortgage payment CAN avoid

*** default and of course when the home is worth less than the mortgage

THEM default and of course when the home is worth less than the mortgage

BEHAVIORS TEND to change

*** PAPERS to change

WHAT DO you see

*** WHEN you see

*** WELL I'M VERY MUCH CONCERNED IF home prices DECLINE FROM HERE GOING to THE REASON IS THAT

ALL KINDS FOR IMAGE FEATURE AND THIS home prices *** *** *** *** to *** *** CLINTON AND

I don't think THEY ARE going to IN OTHER WORDS THEY SEEM TO BE BOTTOMING OUT

*** don't think *** THEY'RE going to *** *** *** *** HAVE RESISTED THE BOMBING OF

the REASON I AM IS THAT DURING 2005 AND 2006 AS I RECALL

the NATION AND TO TURN 2005 AM TOO FAR-FETCHED FOR THE WHOLE TO

THEY WERE EIGHT MILLION home purchases

LET THE MEN IN home purchases

WITH SO CALLED CONVENTIONAL CONFORMING

THE FOCAL CONVENTIONAL CAN FORM

MORTGAGES WITH the twenty percent

OR IN the twenty percent

DOWN PAYMENT THAT down PAYMENT IS GONE and WE HAVE THIS VERY large block of *** *** A HOMEOWNERS WHO ARE RIGHT on the edge of tilting down into that *** *** *** UNDERWATER CATEGORY

*** *** *** down *** AND THAT, and *** SCOTT AND FOR large block of YOU IN THE HOME OWNERS WHO WRITE on the edge of tilting down into that AND A ONE OF OUR

fortunately the evidence suggests that the vast majority AS I WAS IMPLYING

fortunately the evidence suggests that the vast majority *** OF CURRENT VERSION

of these types of HOME OWNERS THAT IS THOSE WITH THE standard conventional mortgages

of these types of *** *** *** HOMEOWNERS AND SO FORTH standard conventional mortgages

AH DO CONTINUE TO PAY on *** *** THEIR MORTGAGES even if the value of the homes is below *** *** THEIR the market PRICE

AND CONTINUE THE PAGE AND on THE NORWEGIAN SHIP TO even if the value of the homes is below THE EVENT, AND the market ***

*** *** *** *** *** ER RATHER WHAT WORRIES ME PARTICULARLY

AND THE AND A WILL WE SHOULD HAVE TO SHOW HE

is THAT THERE IS A VERY large block

is *** THE ENGINE FOR YOUR large block

that will be thrown *** ON THE market

that will be thrown BALL AND A market

*** MEET people STATING to foreclose

THAT THE people STARTED to foreclose

*** if prices go down significantly from here

TO if prices go down significantly from here

but let me move on I WANT TO talk about the DEFICIT AND I ALSO WANT to talk about taxes

but let me move on *** *** AND talk about the *** DEATHS IN A SLOW to talk about taxes

here are the deficit projections from the president said twenty eleven

here are the deficit projections from the president said twenty eleven a budget and *** numbers are frankly staggering if you look AT the deficit for twenty ten 1 65 TRILLON

a budget and THE numbers are frankly staggering if you look OF the deficit for twenty ten 1 56 TRILLION

and THROUGH twenty fifteen

and TWO twenty fifteen

they ESTIMATED it comes down with seven point *** SEVEN HUNDRED FIFTY one point nine billion

they ESTIMATE it comes down with seven point EN SENSE EVIDENCE OF one point nine billion

AH

OF

how serious is this secretary paulson

how serious is this secretary paulson

assuming also THAT TEN YEAR projections are often wrong assuming also *** THE TENURE projections are often wrong

OH I   just HAVE no doubt

*** *** just HAD no doubt

*** that IT IS by far

THAT that HE HAS by far

the most serious LONG TERM   challenge

the most serious *** LONG-TERM challenge

we as a   nation FACE

we as a   nation FIX

all these other issues

all these other issues

***   *** economic issues are minor compared to that

FROM OUR economic issues are minor compared to that

that the THE that *** EH/S>

that the UP that THE CAN

and IS A generational issue

and *** SHOULD generational issue

because ITS

because IT

THERE IS no way WE ARE going to UM to

*** GIVES no way *** WE'RE going to LEAD to

deal effectively *** with DE deficit

deal effectively WITH with THE deficit

without AH AH reforming the entitlement programs AH

without THE TO reforming the entitlement programs ***

*** AH medicare *** medicaid social security

TO THE medicare AND medicaid social security

and *** it doesn't have to be *** A crisis and THAT it doesn't have to be PAID THE crisis

THIS IS something that can be handled

IS TO something that can be handled

*** BUT WA one of the things THAT i I I talk about ON MY BOOK AND ONE of THE LESSONS THAT just HIT ME right between the eyes IN BEING IN WASHINTON

TO BUY A one of the things *** i LIKE TO talk about *** *** *** A BLOCK of OLD LICENSE IS just *** THE right between the eyes *** *** AND WASHINGTON

is *** THAT'S very very difficult to get congress to act

is IT'S A very very difficult to get congress to act

ON ANYTHING THAT is BIG and difficult and controversial

OF THE THING is PAID and difficult and controversial

if THERE IS not an immediate crisis

if *** THERE'S not an immediate crisis

and *** SO THIS

and THE SAGA IS

SO WHAT IS GOING TO take

*** STILL ONE IT'S GONNA take

to *** *** to GET LEADERS ON BOTH sides to come together AND deal with *** *** THIS i think is a huge question

to THE TUNE to THE CAVALIERS OF ALL sides to come together TO deal with THE SIDE THAT i think is a huge question

and *** DR GREENSPAN larry summers ONE of THE PRESIDENT'S top economic ADVISORS UH has said in the past HE'S ASKED a very provocative question which is

and EVERY SPAM AND larry summers *** of *** HIS top economic ADVISERS BUT has said in the past BABIES AS a very provocative question which is

how CAN THE LONG the world's biggest *** BORROWER remain the world's biggest POWER

how *** LONG CAN the world's biggest BAR WAR remain the world's biggest THAN

UH NOT INDEFINITELY BECAUSE THERE IS no doubt

*** *** SMART AND DEFINITELY POSES no doubt that if *** united states CONTINUES DOWN THE road THAT HANKY IS BEEN CORRECTLY INDENTIFYING UH WE ARE GOING to FIND THAT OUR ABLILITY TO BORROW

that if THE united states CAN TWO NEWS, road *** *** *** *** *** *** *** *** *** *** to *** WHOM TO THEM CORRECTLY IDENTIFY

*** *** *** *** *** *** is GOING to GET RESTRAINED BECAUSE throughout OUR history *** *** *** WE HAVE ALWAYS MAINTAINED

FIND ACALLER OF DILUTED THE BALL is CORNERED to RESTRAIN THE COURTS throughout THE history OF THE OF HOLY MEN AND THE

*** A CAPITAL CUSHION A CUSHION BETWEEN

CAPITOL PUSH AND PUSH AND 28 OF

OUR BORROWING capacity ON one END and OUR level of DEBT ON THE OTHER that is beginning to SHRINK and if we get to the POINT WHERE ARE WE HAVING DIFFICULTY SELLING OUR security OUR TRESURY issues

THE RULING capacity THAN one *** and THE level of *** THAT COMING OVER that is beginning to ENSURE and if we get to the *** *** POLLING ROOM, DIFFICULT RATIO OF EACH security AND TREASURY issues

UH THEN interest rates begin to move

OF THE interest rates begin to move

and our ability to move *** *** INTERNATIONALLY

and our ability to move INTO NATIONAL AND

to essentially BE the MAYOR currency the MAYOR economy UH the MAYOR economic power in the world IS significantly DIMINISHED

to essentially THE the MAJOR currency the MAJOR economy OF the MAJOR economic power in the world THAN significantly DEMAND

history tells US THAT great powers

history tells OF THE great powers

WHEN THEY HAVE GOTTEN into very significant fiscal problems HAVE CEASED TO BE great powers

*** WOULD CUT THEM into very significant fiscal problems *** BUT SINCE THE great powers the *** part of the FIX HERE ACCORDING TO THE budget HAS TO do with the issue of taxes

the TWO part of the *** STATE'S YOU RECORDED A budget *** AS do with the issue of taxes

THIS IS HOW THE wall street journal put IT in A headline on tuesday and that is THAT the wealthy face a tax increase those bush ERA tax cuts are going to be allowed to expire by this administration

*** HAS HAD A wall street journal put *** in THE headline on tuesday and that is *** the wealthy face a tax increase those bush *** tax cuts are going to be allowed to expire by this administration

SECREATARY PAULSON IS THAT A BAD IDEA?

SECRETARY PAUL SINGH IS AT THAT IDEA

HERE IS HOW I look at TAXES

IS THERE STILL A look at TEXAS

I BELIEVE THAT what WE NEED IS broad based tax reform

A TE LLEVE what THE GUY HAS broad based tax reform and the kind of tax reform

and the kind of tax reform

WHERE there *** *** DOSEN'T discourage investments AND savings OR INCENTIVES for those RI right now

WERE there THAT IT DOESN'T discourage investments *** savings SERVICE CENTERS for those WERE right now

*** we have A tax system *** *** *** is biased TOWARDS consumption ITS AH

THAT we have THE tax system THAT TO THAT is biased OR consumption AND THAT'S

*** *** and AND AND WE AS we *** AS A people *** SAVE TO little AH invest TO little BORROW TO much

THE THING and THEN WE'LL SEE WHAT we ALL USE OUR people SAY IT'S TOO little THAT invest TOO little ABOUT TOO much

AH so i *** i WILL like to see *** *** WHOLESALE broad based tax reform

TO so i THOUGHT i WOULD like to see THE WHOLE SCENARIO broad based tax reform and I i think that's *** that's clearly GOTTEN

and *** i think that's THE that's clearly OF

my question IF WEATHER the BUSHES tax *** CUT EXPIRING WAS a bad idea

my question IS WHETHER the BUSH tax CUTS EXPIRE AND IS a bad idea

*** WELL I GOT TO say anything right now that IS going to

WHERE ALL THE DATA OF, say anything right now that HE'S going to

AH that *** *** IS GOING TO AFFECT the *** A a tax increase

THE that HAS GONE ON THE FACT OF the EIGHT AND a tax increase

IS GOT TO BE

TO SEE THIS COMEDY

AH IS GOT TO BE QUESTIONED

*** OF THE DIS, THE QUESTION

AND an expiring tax cut is a tax INCREASE, BUT I AM going beyond that

TO an expiring tax cut is a tax *** *** INCREASE ON going beyond that because i really do believe that we are going to need

because i really do believe that we are going to need

*** A to take a different approach to a number of THINGS taxes being one of *** *** THEM HOUSING policies BEING another

THE THING to take a different approach to a number of SIGNS taxes being one of THE MY HOUSE AND policies TO another

DR GREENSPAN THE TAX CUTS UH UH I AGREE WITH WHAT HANK IS SAYING I THINK the THING THAT DISTURBED ME MOST IN THE last week OR TWO WAS WHEN THE discussion was involved IN I BELIEVE in the SENATE

*** WITH A RAISED IN THE LICENSE THROUGH A DEALER GARDEN FROM HOUSTON AND SENTENCE INTO the *** *** *** *** STORM MOVED FROM last week *** *** TO ORANGE LAND discussion was involved *** *** *** in the EMISSION

ON the issue of *** FORMING A COMMISSION A CONGRESSIONALLY AUTHORIZED COMMISSION AS i read IT

*** the issue of FORM AND COMMISSIONED THE CONGRESSIONAL SCHOOL HAS MENTIONED THAT i read THAT there was a NINETY SEVEN to nothing VOTE

there was a *** 97 to nothing FULL

to exclude social security

to exclude social security

from the deliberations of THAT commission

from the deliberations of THE commission

that SAID TO ME that we've GOTTEN TO the POINT IN this country

that *** *** CERTAIN that we've GOT INTO the *** POLLING this country

where spending IS UNTOUCHABLE

where spending THE TIME,

AH I have no doubt THAT we have to raise taxes IN ORDER TO CLOSE THIS huge deficit

TRIPLE THE have no doubt TO we have to raise taxes *** *** *** AND CLOTHES huge deficit but we cannot do IT WHOLLY ON THE TAX SIDE BECAUSE

but we cannot do *** *** WHEN HOLY ROOM, TO IMPOSE

that WILL significantly ERODE

that WOULD significantly UNROLLED

the rate of growth in the economy and the tax base

the rate of growth in the economy and the tax base

and THE REVENUES THAT WILL be achieved WILL BE FAR LESS THAN anybody EXPECT

and *** REVENUE SHARE WOULD be achieved *** THE FALL OF SOME anybody THAT

we have to recognize the fact that one of the things that we have to do

we have to recognize the fact that one of the things that we have to do

AS TOUGH as ITS going to be

IS HALF as IT'S going to be

IS THAT benefits WILL HAVE TO BE paired

*** THE benefits TO ½ FOR THE paired

in conjunction with tax increases

in conjunction with tax increases

to resolve THIS VERY SERIOUS

to resolve *** THE ISSUES

LONG TERM BUDGET PROBLEM

*** *** LONG-TERM BUDGET,

IN OUR REMAINING moment SECRETARY PAULSON I HAVE TO ask you about

THERE ARE MANY moment HIS SECRETARY PAUL SOME AND ask you about

financial regulation

financial regulation

about bonuses on wall street

about bonuses on wall street

DO you see *** REAL CHANGES happening on wall street ARE YOU frustrated by the LEVEL OF bonuses WE ARE SEEING

*** you see WE'LL CHANGE IS happening on wall street *** THE frustrated by the *** LOCAL bonuses *** WERE SAFE

WELL YOU ASKED TWO questions and SO

FOR THE SIERRAS TO questions and CEO

AH first LIKE

OF first LIGHT

TH TH THERE IS no doubt that the *** THAT the compensation

*** *** THAT THERE'S no doubt that the FED TO the compensation

ON wall street

OF wall street

I THINK IS OUT of whack BEEN OUT OF WHACK for some time

*** *** THE EDUCATION of whack *** AND A WET for some time

AND I understand why

*** TO understand why

the american people

the american people

are unhappy because ON OUR system *** we expect those WHO take RISKS TO UH

are unhappy because THE HONOR system WE we expect those THAT take *** *** WRISTS

*** *** *** *** TO TO bare their own losses

AND TWO LUCK THAT ARE REALLY bare their own losses

but i WILL like to see that that *** frustration THAT anger CHANNELED

but i WOULD like to see that that THE frustration AND anger CHANNEL

*** *** TOWARD regulatory reform and I i just think that's very very critical AND TO ME

TO WORK FOR regulatory reform and *** i just think that's very very critical SENT IN THE one thing that IS ABSOLY essential *** that we *** we *** GET strong resolution authority so THAT IN the future any type of FINACIAL institution

one thing that HAS ACTUALLY essential IS that we BELIEVE we HAD A strong resolution authority so *** *** the future any type of FINANCIAL institution

*** *** WHEN IT FACES FAILURE

WANTED US TO FACE IS FAMILIAR

*** that THE that that *** is liquidated

THAT that THAT that that DATA is liquidated

and liquidated in a way

and liquidated in a way

IN which the TAX PAYER IS not GOING TO HAVE to COME UP IN AGAIN

TO which the *** TAXPAYER DOES not HAVE TO, UP to *** THE GUY AND

and PROP up or bail out *** *** A financial institution

and POPPED up or bail out THE EIGHTH THE financial institution

ALRIGHT we will LEAVE IT THERE BUT BEFORE WE LET YOU GO HERE IS THE picture of you back IN THE playing days AT dartmouth HERE SO I GOT TO ASK WE HAVE A COUPLE OF football fans

BUT we will *** *** *** *** *** *** ONLY IDENTIFY WHICH AGO HERE'S A picture of you back *** INTO playing days OF dartmouth *** *** *** *** *** *** *** ¥AN ASTHMA ATTACK DOUBLE football fans

*** *** DR GREENSPAN YOU ARE as well super BALL PICKS secretary paulson you first

THAN NO BUT THE GREENSPAN YOUR as well super BOWL PICK secretary paulson you first

WELL I'M GONNA GO with the *** *** AH indiana and PETE MANNING OK VERY DIFFICULT TO GO AGAINST PAYTON EYE MY VIEW as well WE WILL MAKE THAT THE LAST WORD THANK YOU BOTH FOR

*** ROW OF MIGUEL with the AID TO THE indiana and *** THE MAN A BUDGET FOR DEFENSE PROGRAMS THAT IF I DO as well ONLY THAT ALLOWS WHERE THEY GIVE A PHRASE LIKE THOSE EVENTS

WE will continue our discussion WITH SECRETARY paulson

*** will continue our discussion *** DYSENTERY paulson and ASK HIM some questions

and *** ASKING some questions

THE VIEWERS HAVE SUBMITTED VIA email AND TWITTER ITS ON our meet the press take two web extra YOU can also READ AN EXTRACT from his book on the brink inside the race to stop the collapse of the global

*** THAT YOUR SYSTEM OF email *** IT WITH ERICSSON our meet the press take two web extra *** can also *** BE ANSWERED from his book on the brink inside the race to stop the collapse of the global

financial system

financial system

PLUS SO OTHER UPDATE FORM me throughout the week all ON our website *** *** MTPMSNBC com

*** *** ALSO PROMPTED FOR me throughout the week all *** our website INTO THE MSNBC com

AND UP next SARA palin rallies THE TEA party and the FORTY FIRST gop senator IS SWORN IN

*** AT next SARAH palin rallies THAT SI party and the *** 41ST gop senator *** THIS MORNING

HOW WILL IT ALL impact the obama agenda AND the 2010 ELECTIONS A ROUND TABLE ED GILLESPIE and DEE DEE MYERS

*** *** *** LITTLE impact the obama agenda IN the TWENTY TEN HOW MUCH OF THE MOUNTAIN and LESBIAN THE DEMISE

ONLY ON MEET THE PRESS

OF THE ONLY AND IF

TOTAL Words: 3286 Correct: 1831 Errors: 1685

TOTAL Percent correct = 55.72% Error = 51.28% Accuracy = 48.72%

TOTAL Insertions: 230 Deletions: 316 Substitutions: 1139

APPENDIX B: TEST SEARCH SAMPLE DATA

fuzzySearchWord	*fuzzyPhonemeSearch*	*M_MetaPhone1*	*P_PSearched*	*fuzzyPDSLeftVal*	*fuzzyPDSRightVal*	*fuzzyPhonemeTotalFreq*	*M_TotWordsFound*	*P_TotalPFound*	*W_TotalFreq*	*timeStamp*	*GroupSearchGUID*	*%CFuzzy*	*%CMeta*	*%CPhoneme*	*%CWord*
greenspan	%g r iy n s p ae%	KRNS	g r iy n s p ae n	0	0	47.0	60.0	47.0	47.0	10/13/2010 13:37:03	eaf32d46-0e84-40b2-93f3-f6d86fcf260c	0.00%	27.66%	0.00%	0.00%
economy	%ih k aa n __ m%	AKNM	ih k aa n ax m iy	0	0	199.0	410.0	172.0	244.0	10/13/2010 13:45:08	8ac87835-8cc6-441a-bf06-4ad1143c6dc9	-18.44%	68.03%	-29.51%	0.00%
jobless	%jh aa b l __%	JPLS	jh aa b l ax s	0	0	82.0	68.0	63.0	68.0	10/13/2010 15:25:29	785c0fce-b0bb-4ba9-b048-1f6de5d5714d	20.59%	0.00%	-7.35%	0.00%
unempl	%ah n ih	ANMP	ah n	0	0	117.0	124.0	117.0	118.0	10/1	d08f780e	-0.	5.0	-0.	0.0

oyment	m p l oy m __ n%		ih m p l oy m ax n t							3/2010 15:32:41	-f8f2-4f1f-a31f-13ccbac24540	85%	8%	85%	0%
job	%jh aa%	JP	jh aa b	0	0	446.0	119.0	412.0	412.0	10/13/2010 16:19:27	a8e6b37d-07a6-4612-af3e-51a1008a6bbc	8.25%	-71.12%	0.00%	0.00%
president	%p r eh z __ d __ n%	PRST	p r eh z ax d ax n t	0	0	391.0	391.0	4.0	347.0	10/13/2010 18:03:57	f17babdc-7221-46ee-a0bd-b9abf4637033	12.68%	12.68%	-98.85%	0.00%
banking	%b ae ng k ih%	PNKN	b ae ng k ih ng	0	0	11.0	9.0	9.0	9.0	10/13/2010 18:08:54	0dc3652b-dcc3-4cd4-a37a-2101ea2f5136	22.22%	0.00%	0.00%	0.00%
econ	%iy k	AKN	iy k	0	0	114.0	410.	0.0	0.0	10	4573	W	W	W	W

omical	__ n aa m ih k __%	M	ax n aa m ih k ax l				0			/13/2010 18:29:29	e523-b96b-4055-8bfb-3d4982f12564	ordSNoHits	ordSNoHits	ordSNoHits	ordSNoHits
disaster	%d ih z ae s t __%	TSST	d ih z ae s t ax r	0	0	38.0	38.0	10.0	38.0	10/13/2010 18:38:59	41017386-148e-4613-9bdf-a88fe1f87102	0.00%	0.00%	-73.68%	0.00%
dollars	%d aa l __ r%	TLRS	d aa l ax r z	0	0	1.0	26.0	1.0	21.0	10/13/2010 19:01:17	8c963e03-2148-4293-822d-7912292b1cec	-95.24%	23.81%	-95.24%	0.00%
mortgages	%m ao r g ih jh ih%	MRTK	m m ao ao ao ao ao ao ao	0	0	9.0	21.0	0.0	6.0	10/17/2010 2:15:31	34354332-5c50-4896-ad5a-	50.00%	250.00%	-100.00%	0.00%

			ao ao r g ih ih ih ih ih ih jh jh jh jh jh ih ih ih ih z								f07d6b5892ad				
mortgages	%m ao r g ih j%	MRTK	m ao r g ih jh ih z	0	4	21.0	21.0	9.0	6.0	10/17/2010 2:18:50	1505c384-a4ee-4257-b4db-2ea13f28da08	250.00%	250.00%	50.00%	0.00%
mortgages	%m ao r g i%	MRTK	m ao r g ih jh ih z	0	7	21.0	21.0	9.0	6.0	10/17/2010 2:21:15	618e2f06-ff80-4bd4-b86d-1941a30facba	250.00%	250.00%	50.00%	0.00%
mortgages	%m ao r g%	MRTK	m ao r g	0	9	28.0	21.0	9.0	6.0	10/17/	be80dbc5-	366.67	250.	50.0	0.0

			ihjhih z							2010 2:25:09	981d-4a29-a783-2da71cb02fbe	%	00%	%	%
mortgages	%mao r%	MRTK	maor gihjhih z	0	10	398.0	21.0	9.0	6.0	10/17/2010 2:29:49	6ca63e17-9e84-4b10-bfad-aedeb694f74d	6533.33%	250.00%	50.00%	0.00%
mortgages	%mao r gih jhih%	MRTK	maor gihjhih z	0	0	9.0	21.0	9.0	6.0	10/18/2010 15:29:16	130e3c99-9ee2-4edf-8b92-faef1a78a79d	50.00%	250.00%	50.00%	0.00%
mortgages	%mao r gih j%	MRTK	maor gihjhih z	0	4	21.0	21.0	9.0	6.0	10/18/2010 15:34:03	b8b3fbce-f3b6-414a-ad80-85c2706a9e2a	250.00%	250.00%	50.00%	0.00%

mortgages	%mao rg%	MRTK	m ao r g ih jh ih z	0	9	28.0	21.0	9.0	6.0	10/18/2010 15:36:02	3385cf4c-bd14-468d-a3a2-7bb372c5ab33	366.67%	250.00%	50.00%	0.00%
mortgages	%mao r%	MRTK	m ao r g ih jh ih z	0	10	398.0	21.0	9.0	6.0	10/18/2010 15:40:39	ede09b5d-9ef6-43a3-be82-bcc86611e374	6533.33%	250.00%	50.00%	0.00%
federal	%feh%	FTRL	f eh d ax r ax l	0	10	331.0	58.0	1.0	56.0	10/18/2010 15:48:37	d94fd6f5-bd3a-4e54-ba89-e078bec013e2	491.07%	3.57%	-98.21%	0.00%
federal	%feh d __ r __%	FTRL	f eh d ax r ax l	0	0	3.0	58.0	1.0	56.0	10/18/2010 15:51:54	b9341018-f3b3-4154-bee4-	-94.64%	3.57%	-98.21%	0.00%

federal	%f ehd __ __%	FTRL	f ehd ax r ax l	0	4	47.0	58.0	1.0	56.0	10/18/2010 15:58:53	6c97493ffcf4835211f8-87a8-40f4-84c3-c02e5e459cd3	-16.07%	3.57%	-98.21%	0.00%
federal	%f ehd%	FTRL	f ehd ax r ax l	0	7	100.0	58.0	1.0	56.0	10/18/2010 16:02:50	e3b491e7-1b96-4c36-af7d-6797d3ebdeba	78.57%	3.57%	-98.21%	0.00%
federal	%d __ r __%	FTRL	f ehd ax r ax l	5	0	160.0	58.0	1.0	56.0	10/18/2010 16:10:59	734d5818-e28a-4377-9d3c-787f0462638f	185.71%	3.57%	-98.21%	0.00%
fed	%f ehd __ __ r __%	FTRL	f ehd ax ax r	0	0	12.0	58.0	12.0	56.0	10/18/2010	2e18acb0-8de6-	-78.57%	3.57%	-78.57%	0.00%

			ax l							16:14:36	4fad-b1c0-83cd93d422a4				
president	%p r eh z __ d __ n%	PRST	p r eh z ax d ax n t	0	0	355.0	391.0	4.0	347.0	10/19/2010 15:41:44	d05033ff-2724-43f1-bc6a-b7ff9394c0b8	2.31%	12.68%	-98.85%	0.00%
president	%p r eh z __ d%	PRST	p r eh z ax d ax n t	0	4	363.0	391.0	4.0	347.0	10/19/2010 16:12:28	18286eff-859c-4039-8950-63f90e73e29e	4.61%	12.68%	-98.85%	0.00%
president	%p r eh z __%	PRST	p r eh z ax d ax n t	0	7	482.0	391.0	4.0	347.0	10/19/2010 16:34:13	57b799cb-cb66-48bd-a636-8264c58c08d3	38.90%	12.68%	-98.85%	0.00%
president	%p r eh z%	PRST	p r eh z	0	10	482.0	391.0	1.0	347.0	10/19/	62e02ed0-	38.90	12.6	-99.7	0.00

			ax ax d ax n t							2010 17:54:25	61db-4d26-b4b2-c24970e31c55	%	8%	1%	%
president	%eh z __%	PRST	p r eh z ax d ax n t	4	10	524.0	391.0	4.0	347.0	10/19/2010 17:59:13	306042bd-5738-433c-aa47-eb5bfd23a76f	51.01%	12.68%	-98.85%	0.00%
president	%eh z __ d __ n%	PRST	p r eh z ax ax d ax n t	4	0	361.0	391.0	1.0	347.0	10/19/2010 18:04:43	a7528f25-256b-48a2-a790-f34aab0627f8	4.03%	12.68%	-99.71%	0.00%
president	%z __ d __ n%	PRST	p r eh z ax d ax n t	7	0	370.0	391.0	4.0	347.0	10/19/2010 18:07:41	51e8d423-9105-4f67-aae6-5c0f7926	6.63%	12.68%	-98.85%	0.00%

president	%x d__n%	PRST	p r eh z ax d ax n t	10	0	79.0	391.0	4.0	347.0	10/19/2010 18:16:01	af311a1d517a-fda0-4381-9187-5c80ec654eca	-77.23%	12.68%	-98.85%	0.00%

APPENDIX C: METAPHONE ENCODING SAMPLE

WORD_GUID CONVERSATION_GUID CONTEXT_GUID WORD PHONETIC_KEY1 PHONETIC_KEY2 WORD_POSITION CHAR_POSITION TIME_STAMP

0c46671f-b907-470f-85d8-dfc6020d498d C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A the 0 T 15 78 2010-05-14 19:14:15.110

29895217-4f15-4588-a837-74910366270c C35A9F9C-080F-48C3-BF0B 23 126 2010-05-14 19:14:15.110-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A years ARS NULL 13 68 2010-05-14 19:14:15.110

2db371c2-bc6d-44cd-ac95-dac7eeda4ea2 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A are AR NULL 6 32 2010-05-14 19:14:15.110

312b0f24-4fab-4ca6-864c-735146205de2 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A irons ARNS NULL 24 130 2010-05-14 19:14:15.110

339060e3-f3ac-4a4c-918d-30a80f37f24d C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A head HT NULL 16 82 2010-05-14 19:14:15.110

3e649941-5bbb-4a9d-b25b-19c5d87478a8 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A his HS NULL

48eb870f-9a48-450a-9a70-6761a6932d5d C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A professors PRFS NULL 4 17 2010-05-14 19:14:15.110

65767a59-6daf-46e9-97c1-98522d191492 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A that 0T TT 8 42 2010-05-14 19:14:15.110

6c4d4a4e-c8cc-4a38-bc85-ca067a65991f C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A and ANT NULL 5 28 2010-05-14 19:14:15.110

713a83a8-965e-4c8a-bd83-82df202c6eb0 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A program PRKR NULL 21 115 2010-05-14 19:14:15.110

72048a99-074f-4811-90c0-fd1094ef62dd C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A on AN NULL 2 9 2010-05-14 19:14:15.110

7397f5d1-25fe-4ba7-add4-86fffdf56465 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A some SM NULL 3 12 2010-05-14 19:14:15.110

763447e1-bba0-48be-b1fb-db16760ad5dc C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A the 0 T 29 154 2010-05-14 19:14:15.110

810d396e-6008-4236-9542-d26e7523ba6f C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A was AS FS 14 74 2010-05-14 19:14:15.110

999b63f9-9f3a-45d0-8c0f-31b0eb5e22c9 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A many MN NULL 12 63 2010-05-14 19:14:15.110

9e2f517a-809a-4e82-b15a-aa98db7d06d9 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A for FR NULL 11 59 2010-05-14 19:14:15.110

a12bc49e-f285-4d96-bf63-61c70e0a034b C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A undergraduate ANTR NULL 19 94 2010-05-14 19:14:15.110

a925c078-4276-44ea-a001-ed7d2e4ca150 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A titles TTLS NULL 27 144 2010-05-14 19:14:15.110

aadb747a-525d-4556-8683-ef8c130e79ad C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A of AF NULL 28 151 2010-05-14 19:14:15.110

b66fd8a1-2e0c-4135-bc85-637c34191013 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A signs SNS SKNS 7 36 2010-05-14 19:14:15.110

bc1bf518-a524-4af5-af0c-b4313b055242 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A in AN NULL 22 123 2010-05-14 19:14:15.110

c54d827f-4b1c-422f-9e5b-068d45887964 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A and ANT NULL 1 5 2010-05-14 19:14:15.110

cefc6d48-1c89-4166-b689-a647dcda709a C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A of AF NULL 17 87 2010-05-14 19:14:15.110

d0d46273-3d71-4216-a4a3-bbe9f7971147 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A the 0 T 18 90 2010-05-14 19:14:15.110

dd65c572-b9f1-4079-b2c5-6d398b324f18 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A degree TKR NULL 20 108 2010-05-14 19:14:15.110

e3ed3755-0735-4e2e-b21a-2bb8cd63c724 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A the 0 T 26 140 2010-05-14 19:14:15.110

ea2f3fe0-ba01-4143-93c9-e5469acb4b95 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A Harvard HRFR NULL 9 47 2010-05-14 19:14:15.110

f033e1c0-4830-415f-99cc-31a3a4eae8e9 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A and ANT NULL 25 136 2010-05-14 19:14:15.110

f047cdfc-1bee-474b-a9aa-304ff95650c7 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A Does TS NULL 0 0 2010-05-14 19:14:15.110

fa71118f-1a03-4b51-9078-8afc816b6774 C35A9F9C-080F-48C3-BF0B-2EB0B4E1E400 AB8A570F-D4CE-4510-9E03-E9EA6CB87F5A and ANT NULL 10 55 2010-05-14 19:14:15.110

00f717f3-6d91-4b6d-a01e-b189e6c06911 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC I A NULL 3 20 2010-05-14 19:14:25.180

0914df05-4fb7-40c3-8b3d-dd9390ae36a3 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC just JST AST 56 317 2010-05-14 19:14:25.180

0ab4c2cd-c35e-462b-841d-b4a13dfe6e95 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC of AF NULL 25 138 2010-05-14 19:14:25.180

0da6f085-a590-4eea-a54e-c9aa604054e2 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC well AL FL 32 176 2010-05-14 19:14:25.180

10da30b4-0d0a-4d3a-afa5-ec17ad3881d2 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC its ATS NULL 26 141 2010-05-14 19:14:25.180

11bae084-6c9a-492c-bb5a-c9902eb78818 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC time TM NULL 51 285 2010-05-14 19:14:25.180

13da4616-665e-47ce-863e-74160220ec42 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC program PRKR NULL 46 253 2010-05-14 19:14:25.180

145950f8-5e92-4e16-ad53-9f1392416b4a 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC scientist SNTS NULL 24 128 2010-05-14 19:14:25.180

17d56cf0-9fd6-4058-9282-e5ffebe9e374 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC for FR NULL 13 66 2010-05-14 19:14:25.180

fb704037-a54c-47e5-b272-dc138bbbdcb8 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC like LK NULL 5 24 2010-05-14 19:14:25.180

2aa902b8-5529-4e80-9d78-be8e205d2fe4 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC basis PSS NULL 35 188 2010-05-14 19:14:25.180

2b81f400-04c3-4e4f-94c8-db60fefacafd 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC treated TRTT NULL 40 225 2010-05-14 19:14:25.180

30218c45-8262-4281-b748-a16965e0663d 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC in AN NULL 22 116 2010-05-14 19:14:25.180

3033630f-a2c3-4b37-a002-e5b52e0b8a79 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC of AF NULL 36 194 2010-05-14 19:14:25.180

32bda410-bec2-4458-bf5d-7cc45737a4f7 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC d T NULL 4 22 2010-05-14 19:14:25.180

35cc41e0-54f4-47be-8fa1-d9975c8f0214 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC condition KNTX NULL 61 347 2010-05-14 19:14:25.180

3bb44b68-eea6-44cf-a95e-1c4eccab2af1 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC have HF NULL 48 268 2010-05-14 19:14:25.180

4097007d-6a75-45d2-9a34-40cf129bfca6 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC computer KMPT NULL 23 119 2010-05-14 19:14:25.180

468e48d1-2573-4f3b-befe-9f7700ee6887 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC getting KTNK NULL 53 293 2010-05-14 19:14:25.180

4b6d13f7-3eae-49b3-a4a7-8dd748a76d52 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC the 0 T 49 273 2010-05-14 19:14:25.180

5a137793-c95a-40a6-b448-bfae8f822667 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC in AN NULL 29 160 2010-05-14 19:14:25.180

5cb2bfc2-d181-4fa0-9c10-fb8a3477ad6b 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC the 0 T 14 70 2010-05-14 19:14:25.180

5d1a606a-c400-42c2-aa19-0eac0e5b8f1f 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC should XLT NULL 47 261 2010-05-14 19:14:25.180

5f19663b-eae7-4642-b3ef-3a10b6ce98aa 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC economics AKNM NULL 30 163 2010-05-14 19:14:25.180

5fb7a873-5212-4bd2-a356-750c32e5dda6 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC the 0 T 17 86 2010-05-14 19:14:25.180

60070659-0c4a-49d2-bc17-75ed1ba36e29 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC specialization SPSL SPXL 39 210 2010-05-14 19:14:25.180

603b15a7-0031-47ec-9ec0-82afd6618959 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC straightSTRT NULL 10 47 2010-05-14 19:14:25.180

698a9239-3a62-4e67-971f-6112a8d11ede 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC set ST NULL 7 32 2010-05-14 19:14:25.180

71b8c5c1-2441-49c3-a024-170e0bc3d869 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC the 0 T 34 184 2010-05-14 19:14:25.180

766a56f0-a549-4f95-9288-716e7e2e57e7 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC to T NULL 6 29 2010-05-14 19:14:25.180

75cf7b8f-9bda-41dd-88d9-a0737173a74e 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC easiest ASST NULL 50 277 2010-05-14 19:14:25.180

79d9cd6a-0811-44bd-b1fb-3268210936c1 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC Harvard HRFR NULL 0 0 2010-05-14 19:14:25.180

8e8ff09a-2fd7-4b5f-bdab-d68e715a23bd 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC of AF NULL 52 290 2010-05-14 19:14:25.180

8f4d5673-c152-45d9-8032-78b380dac3fe 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC knowledge NLJ NULL 21 106 2010-05-14 19:14:25.180

95268116-0929-4736-9714-bc0d0d0143a2 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC with A0 FT 41 233 2010-05-14 19:14:25.180

9a1e8dfe-a742-4717-8494-15a5528c9a6c 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC of AF NULL 20 103 2010-05-14 19:14:25.180

9a3037f0-585b-43ff-a9d0-c08c9aa1e671 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC Bruce PRS NULL 19 97 2010-05-14 19:14:25.180

9e3cfdfc-3123-4193-a469-5c18fb1a7128 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC record RKRT NULL 9 40 2010-05-14 19:14:25.180

9ef8f28d-cad6-4b8e-8bb8-bd763bc81818 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC person PRSN NULL 12 59 2010-05-14 19:14:25.180

a0c3c57c-5435-4ca6-9e40-305733b8dbf0 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC the 0 T 8 36 2010-05-14 19:14:25.180

a2e1e335-3541-4087-94c6-3bb6d16649e4 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC an AN NULL 42 238 2010-05-14 19:14:25.180

a96b640a-5103-4930-b760-65ab6ca19bcd 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC Harvard HRFR NULL 55 309 2010-05-14 19:14:25.180

ac748d11-2db5-4065-93fd-e1e04c2dfe43 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC out AT NULL 44 245 2010-05-14 19:14:25.180

adade126-7258-42ec-969a-1fa340adf5e7 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC background PKKR NULL 28 149 2010-05-14 19:14:25.180

b256d814-63a4-4806-affe-ad95e943272b 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC financial FNNS FNNX 60 337 2010-05-14 19:14:25.180

b2695f85-65c9-421f-89a1-85480ac5d543 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC opt APT NULL 43 241 2010-05-14 19:14:25.180

b55a15f0-6e25-455a-9a7d-d2e9a9a6062c 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC the 0 T 37 197 2010-05-14 19:14:25.180

ba4fc4c0-9b55-44ff-8b19-a171f6901c1a 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC that 0T TT 16 81 2010-05-14 19:14:25.180

c23aa0ca-5918-4b7c-9d81-96e9c4904b4f 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC economic AKNM NULL 38 201 2010-05-14 19:14:25.180

c23efb95-bd72-406f-876f-975c822dc9b3 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC on AN NULL 33 181 2010-05-14 19:14:25.180

c8798609-4c09-4f81-a6be-dfb27b568d31 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC the 0 T 45 249 2010-05-14 19:14:25.180

c8d0c79c-f7ef-422a-be08-408eb881a628 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC record RKRT NULL 15 74 2010-05-14 19:14:25.180

cb1871d9-0a63-4f9c-92a9-9d893862b2dc 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC as AS NULL 31 173 2010-05-14 19:14:25.180

ce5ae54c-5a9b-4ee0-abb5-3e1d4181f537 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC through 0R TR 54 301 2010-05-14 19:14:25.180

ce1e5817-c1ff-4ad7-83b3-26bc051a7b29 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC of AF NULL 58 330 2010-05-14 19:14:25.180

cc7a567b-f77c-42dc-90ae-65f2c6b7d5eb 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC and ANT NULL 2 16 2010-05-14 19:14:25.180

ddf22921-ccfc-4b2e-b60d-4d99c07e8ba0 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC own AN NULL 27 145 2010-05-14 19:14:25.180

e895e223-a8c6-450c-89d4-83b1051d0cd1 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC college KLJ KLK 1 8 2010-05-14 19:14:25.180

f15a48c5-08af-439f-a7c8-ca603af73e61 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC on AN NULL 11 56 2010-05-14 19:14:25.180

f51e197b-5754-4ceb-a250-f82656a85ee5 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC her HR NULL 59 333 2010-05-14 19:14:25.180

f816005e-4083-4fd4-a8e2-a188c81c7fcf 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC person PRSN NULL 18 90 2010-05-14 19:14:25.180

fb502c49-261a-4dfd-ace9-8a4609952188 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F 27018EDD-F6D4-4829-A58E-7A1F134174AC because PKS NULL 57 322 2010-05-14 19:14:25.180

0134ad70-ceab-4da3-a3ec-5a19fb1c2772 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F A4163D37-1030-4DE9-8570-162063C2EA66 students STTN NULL 68 357 2010-05-14 19:14:34.087

0057e007-347c-4be2-8c28-b25fc032f9b8 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F A4163D37-1030-4DE9-8570-162063C2EA66 come KM NULL 70 369 2010-05-14 19:14:34.087

046fc069-8d46-4363-9ae9-f9282784e455 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F A4163D37-1030-4DE9-8570-162063C2EA66 s S NULL 61 316 2010-05-14 19:14:34.087

05370cf9-9f87-4748-b9f9-febe4f20cb59 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F A4163D37-1030-4DE9-8570-162063C2EA66 easy AS NULL 26 144 2010-05-14 19:14:34.087

061f2253-e251-4982-9591-5af52d5c28eb 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F A4163D37-1030-4DE9-8570-162063C2EA66 one AN NULL 54 288 2010-05-14 19:14:34.087

070f78ce-701d-4cf2-9afb-98b4d87aa469 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F A4163D37-1030-4DE9-8570-162063C2EA66 back PK NULL 47 255 2010-05-14 19:14:34.087

09eadfac-356a-4700-bc29-eece4e2844b1 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F A4163D37-1030-4DE9-8570-162063C2EA66 to T NULL 48 260 2010-05-14 19:14:34.087

0e3c4761-aa76-4bf1-90a7-448b74b43935 2BB515DD-DF43-4BCC-AA4D-A2520DCBDD4F A4163D37-1030-4DE9-8570-162063C2EA66 Linda LNT NULL 6 35 2010-05-14 19:14:34.087

APPENDIX D: SEARCH RESULTS DB DETAIL SAMPLE

fuzzySearchWord fuzzyPhonemeSearch M_MetaPhone1 P_PSearched
fuzzyPDSLeftVal fuzzyPDSRightVal fuzzyPhonemeTotalFreq
M_TotWordsFound P_TotalPFound W_TotalFreq timeStamp
GroupSearchGUID
republicans %ih p ah b l ih k __ n% RPPL r ih p ah b l ih k ax n z2 0 9
80 4 41 2010-10-19 19:53:50.000 0ae3a03f-80fb-4c5f-9628-
255e7a08cca3
banking %b ae ng k ih% PNKN b ae ng k ih ng0 0 11 9 9
9 2010-10-13 18:08:54.000 0dc3652b-dcc3-4cd4-a37a-2101ea2f5136
mortgages %m ao r g ih jh ih% MRTK m ao r g ih jh ih z 0 0 9 21
9 6 2010-10-18 15:29:16.000 130e3c99-9ee2-4edf-8b92-faef1a78a79d
mortgages %m ao r g ih j% MRTK m ao r g ih jh ih z 0 4 21 21
9 6 2010-10-17 02:18:50.000 1505c384-a4ee-4257-b4db-2ea13f28da08
ph.d.s %p iy ey ch d aa t d iy d aa t eh% FTS p iy ey ch d aa t d iy d aa t eh s 0
0 0 7 0 1 2010-10-19 19:41:59.000 168258de-65e8-4d22-
86f1-266fb1c2a6f1
president %p r eh z __ d% PRST p r eh z ax d ax n t 0 4 363 391
4 347 2010-10-19 16:12:28.000 18286eff-859c-4039-8950-63f90e73e29e
president %x d __ n% PRST p r eh z ax d ax n t 10 0 79 391 4
347 2010-10-19 18:16:01.000 1a1d517a-fda0-4381-9187-5c80ec654eca
federal reserve %f eh d __ r __ l r ih z% FTRL f eh d ax r ax l r ih z er r v 0 4
0 58 0 24 2010-10-19 18:38:39.000 1b2c8839-9361-44e1-af92-
41c2f5648373
republicans %r __ p ah b l __ k __ n% RPPL r ih p ah b l ih k ax n z0 0 72
80 4 41 2010-10-19 20:21:36.000 1d055bf8-a2aa-42e2-941c-
ce70ed8bccf7
former %f ao r m __% FRMR f ao r m ax r 0 0 155 60 3 60
2010-10-19 19:15:10.000 21901594-80e0-4101-a2b1-29ebdb813236
payroll tax holiday %p ey r ow l t ae k s h __ l __ d% PRLT p ey r ow l t ae k s h aa l ax d
ey 0 0 5 0 0 5 2010-10-19 21:12:13.000 29eaec0a-
fbbe-474a-b04f-7043bfb472f0
economics %iy k __ n __ m __ k% AKNM iy k ax n aa m ih k s 0 0
68 410 3 33 2010-10-21 00:06:43.000 2a14d97b-49d6-4ca4-bf70-
9a218722ee1a
federal %f eh d __ r __% FTRL f eh d ax r ax l 0 0 3 58 1 56
2010-10-19 21:06:57.000 2d7f2bf5-75cc-4973-87b0-3b1aee87edae
fed %f eh d __ __ r __% FTRL f eh d ax ax r ax l 0 0 12 58 12
56 2010-10-18 16:14:36.000 2e18acb0-8de6-4fad-b1c0-83cd93d422a4
president %eh z __% PRST p r eh z ax d ax n t 4 10 524 391 4
347 2010-10-19 17:59:13.000 306042bd-5738-433c-aa47-eb5bfd23a76f
mortgages %m ao r g% MRTK m ao r g ih jh ih z 0 9 28 21 9
6 2010-10-18 15:36:02.000 3385cf4c-bd14-468d-a3a2-7bb372c5ab33

federal reserve %__ r __ l r ih z er r%FTRL f eh d ax r ax l r ih z er r v 7 0 2
58 0 24 2010-10-19 18:59:05.000 340ad2ed-1a47-4c90-a292-c4a63197796c
mortgages %m ao r g ih jh ih% MRTK m m ao ao ao ao ao ao ao ao ao r g ih ih ih ih ih ih jh jh jh jh jh ih ih ih ih z 0 0 9 21 0 6 2010-10-17 02:15:31.000 34354332-5c50-4896-ad5a-f07d6b5892ad
twenty six thousand %t w eh n t iy s ih k s th aw z __ n% TNTS t w eh eh n t iy s ih k s th aw z ax n d 0 0 2 0 0 0 2010-10-19 19:35:56.000
3c5d415b-8c4d-4ec4-9e05-f7f1b61df6d4
president %z __ d __ n% PRST p r eh z ax d ax n t 6 0 370 391
4 372 2010-10-28 23:19:10.000 3dacb937-89af-4d34-ab7d-548641062948
obama %__ b ae m a% APM ax ax ax ax ax b ae ae ae m ax 0 0 0
91 0 91 2010-10-22 04:40:30.000 3fa36c95-cd58-4416-bb9c-d9641f709d69
disaster %d ih z ae s t __% TSST d ih z ae s t ax r 0 0 38 38
10 38 2010-10-13 18:38:59.000 41017386-148e-4613-9bdf-a88fe1f87102
economical %iy k __ n aa m ih k __% AKNM iy k ax n aa m ih k ax l 0
0 114 410 0 0 2010-10-13 18:29:29.000 4573e523-b96b-4055-8bfb-3d4982f12564
tim cramer %t ih m k r ae m __% TMKR t ih m k r ae m ax r 0 0 0 58
0 0 2010-10-19 19:22:05.000 4602ff82-06df-4380-8b9c-ef846a4be75d
harvard %h __ r v __ r% HRFR h aa r v ax r d 0 0 6 77 4
76 2010-10-21 00:17:51.000 473366ac-4a72-4869-b617-b0f62029b9f1
federalreserve %f eh d __ r __ l r __ z er r% FTRL f eh d ax r ax l r ax z er r v 0 0
0 58 0 0 2010-10-19 18:41:49.000 48f926fe-b1e4-47a3-a654-81dbb5af7b0d
president %z __ d __ n% PRST p r eh z ax d ax n t 7 0 370 391
4 347 2010-10-19 18:07:41.000 51e8d423-9105-4f67-aae6-5c0f7926af31
president %p r eh z __%PRST p r eh z ax d ax n t 0 7 482 391 4
347 2010-10-19 16:34:13.000 57b799cb-cb66-48bd-a636-8264c58c08d3
mortgages %m ao r g i% MRTK m ao r g ih jh ih z 0 7 21 21 9
6 2010-10-17 02:21:15.000 618e2f06-ff80-4bd4-b86d-1941a30facba
federal reserve %r __ l r ih z er r% FTRL f eh d ax r ax l r ih z er r v 10 0 2
58 0 24 2010-10-19 18:45:37.000 6295219b-dfaa-4b54-a7e9-73cc31da9359
president %p r eh z% PRST p r eh z ax ax d ax n t 0 10 482 391 1
347 2010-10-19 17:54:25.000 62e02ed0-61db-4d26-b4b2-c24970e31c55
mortgages %m ao r% MRTK m ao r g ih jh ih z 0 10 398 21 9
6 2010-10-17 02:29:49.000 6ca63e17-9e84-4b10-bfad-aedeb694f74d
the economist %dh dh __ __ k __ n __ m __ s% 0KNM dh dh ih ih k aa n ax m ih s t 0
0 2 0 0 9 2010-10-19 21:25:16.000 727668d0-f7e8-4761-91b5-564e0045764a

federal %d __ r __% FTRL f eh d ax r ax l 5 0 160 58 1 56 2010-10-18 16:10:59.000 734d5818-e28a-4377-9d3c-787f0462638f
twenty six thousnad %t w eh n t iy s ih k s th aw% TNTS t w eh n t iy s ih k s th aw s n ae d 0 7 2 0 0 0 2010-10-19 19:31:33.000 76dfbfc0-7b54-4420-b598-b9ae72980f3e
jobless %jh aa b l __% JPLS jh aa b l ax s 0 0 82 68 63 68 2010-10-13 15:25:29.000 785c0fce-b0bb-4ba9-b048-1f6de5d5714d
federal reserve %f eh d __ r __ l r ih z er r% FTRL f eh d ax r ax l r ih z er r v 0 0 0 58 0 24 2010-10-19 18:28:23.000 7c658bb8-d615-4774-97eb-8096abfe60fc
federal %f eh d __ __% FTRL f eh d ax r ax l 0 4 47 58 1 56 2010-10-18 15:58:53.000 835211f8-87a8-40f4-84c3-c02e5e459cd3
economy %__ k __ n __ m% AKNM ih k aa n ax m iy 0 0 417 410 172 244 2010-10-20 23:32:16.000 83748a4d-8f60-40b6-86c7-2c3f40a30b4c
tony heyward %t ow n iy h ey w __ r% TNRT t ow n iy h ey w ax r d0 0 1 3 0 1 2010-10-19 19:38:17.000 87f6c121-8a0a-4daa-bdae-59d85649643b
economy %ih k aa n __ m% AKNM ih k aa n ax m iy 0 0 199 410 172 244 2010-10-13 13:45:08.000 8ac87835-8cc6-441a-bf06-4ad1143c6dc9
dollars %d aa l __ r% TLRS d aa l ax r z 0 0 1 26 1 21 2010-10-13 19:01:17.000 8c963e03-2148-4293-822d-7912292b1cec
rapublicans %r aa p ah b l i% RPPL r aa p ah b l ih k ax n z 0 8 0 80 0 0 2010-10-19 20:05:53.000 9bcd6e6b-e59c-4fb8-ada8-7ad9da11bfb5
president %eh z __ d __ n% PRST p r eh z ax ax d ax n t 4 0 361 391 1 347 2010-10-19 18:04:43.000 a7528f25-256b-48a2-a790-f34aab0627f8
job %jh aa% JP jh aa b 0 0 446 119 412 412 2010-10-13 16:19:27.000 a8e6b37d-07a6-4612-af3e-51a1008a6bbc
president %p r eh z __ d __ n% PRST p p r eh eh eh eh eh eh eh eh z ax ax ax ax ax ax d ax ax ax ax n t 0 0 355 391 0 347 2010-10-28 21:18:36.000 a9ec50dc-a5e5-4203-8a1c-c3fd87472976
phd %p iy ey ch d% FT p iy ey ch d iy 0 0 1 98 1 0 2010-10-19 19:45:45.000 ac03e92b-bd1f-43fb-b0dc-2f9ee6855f10
phd %p iy ey ch d% FT p iy ey ch d iy 0 1 1 98 1 0 2010-10-29 05:20:17.000 b89660d6-afb6-47c0-a765-882d63b24517
mortgages %m ao r g ih j% MRTK m ao r g ih jh ih z 0 4 21 21 9 6 2010-10-18 15:34:03.000 b8b3fbce-f3b6-414a-ad80-85c2706a9e2a
federal %f eh d __ r __% FTRL f eh d ax r ax l 0 0 3 58 1 56 2010-10-18 15:51:54.000 b9341018-f3b3-4154-bee4-6c97493ffcf4

bureaucracies %b b y uh r aa k r __ s iy% PRKR b y uh r aa k r ax s iy z 0 0
0 177 1 2 2010-10-19 19:49:09.000 b966bde4-71ed-47e1-81c9-f7a528c917ad
26000 %t w eh n t iy s ih k s th aw z __ n% t w eh eh n t iy s ih k s th aw z ax n d 0
0 2 724 0 0 2010-10-19 19:29:20.000 b9af180e-3233-4317-bc5c-f61f0f64c8a1
federal reserve %f eh d __ r __ l r ih% FTRL f eh d ax r ax l r ih z er r v 0 7
0 58 0 20 2010-10-19 18:40:13.000 bb0bf063-2f19-42a3-98fb-d15c5c958b19
bureaucracies %y uh r aa k r __ s iy% PRKR b b y uh r aa k r ax ax s iy z 2 0
1 177 0 2 2010-10-19 19:51:37.000 bd8f9d3c-5497-4207-8298-7e3c6f876511
mortgages %m ao r g% MRTK m ao r g ih jh ih z 0 9 28 21 9
6 2010-10-17 02:25:09.000 be80dbc5-981d-4a29-a783-2da71cb02fbe
republicans %r ih p ah b l i% RPPL r ih p ah b l ih k ax n z 0 8 10 80
4 41 2010-10-19 19:58:30.000 bec52855-c151-4941-b1c0-166e22944d62
p h ds %p iy ey ch d iy eh% PTS p iy ey ch d iy eh s 0 0 0 13 0
0 2010-10-19 19:43:43.000 c291b0e8-f2ce-4cd9-a07a-7409ab5890e6
obama %__ b ae m a% APM ax b ae m ax 0 0 0 91 5 91
2010-10-22 04:47:10.000 cc178849-ce4f-43c6-a057-1e370643bb0a
tim kramer %t ih m k r ey m __% TMKR t ih m k r ey m ax r 0 0 7 58
2 5 2010-10-19 19:24:20.000 cdf394c1-9b02-4963-bb7c-60e565be91ae
president %p r eh z __ d __ n% PRST p r eh z ax d ax n t 0 0 355 391
4 347 2010-10-19 15:41:44.000 d05033ff-2724-43f1-bc6a-b7ff9394c0b8
unemployment %ah n ih m p l oy m __ n% ANMP ah n ih m p l oy m ax n t 0 0
117 124 117 118 2010-10-13 15:32:41.000 d08f780e-f8f2-4f1f-a31f-13ccbac24540
mortgage %m ao r g __ j% MRTK m ao r g ih jh 0 1 21 21 21
21 2010-10-29 04:07:26.000 d46cde31-1ba5-4fcb-a922-bb69990025d2
mortgages %m ao r g __ jh i% MRTK m ao r g ih jh ih z 0 1 9 21
9 6 2010-10-29 04:11:13.000 d7545c2c-2185-40f4-bd1a-0eba3ac19806
economy %__ k __ n __ m% AKNM ih ih ih ih ih ih ih ih k aa aa aa aa aa aa n ax ax ax ax m iy iy 0 0 417 410 0 244 2010-10-20 23:00:35.000
d76a69d3-7140-41c7-8e5e-22b929046985
federal %f eh% FTRL f eh d ax r ax l 0 10 331 58 1 56 2010-10-18 15:48:37.000 d94fd6f5-bd3a-4e54-ba89-e078bec013e2
federal %f eh d% FTRL f eh d ax r ax l 0 7 100 58 1 56 2010-10-18 16:02:50.000 e3b491e7-1b96-4c36-af7d-6797d3ebdeba
federal reserve %f eh d __ r __ l r ih z er r% FTRL f eh d ax ax r ax l r ih z er r v 0 0
0 58 1 24 2010-10-19 18:36:29.000 e3b66ce1-6876-46a2-b670-9343abc4a289

former chairman %f ao r m __ r ch eh r m __% FRMR f ao r m ax r ch eh r m ax n 0 0 0 60 0 13 2010-10-19 19:12:11.000 e419faf8-b9eb-4d46-9052-ba91c951c687
economic %iy k __ n __ m __% AKNM iy k ax n aa m ih k 0 0 137 410 65 141 2010-10-21 00:13:27.000 e7f90fa0-1275-4a08-8234-e0442c6f40c0
greenspan %g r iy n s p ae% KRNS g r iy n s p ae n 0 0 47 60 47 47 2010-10-13 13:37:03.000 eaf32d46-0e84-40b2-93f3-f6d86fcf260c
mortgages %m ao r% MRTK m ao r g ih jh ih z 0 10 398 21 9 6 2010-10-18 15:40:39.000 ede09b5d-9ef6-43a3-be82-bcc86611e374
economist %__ k __ n __ m __ s% AKNM ih k aa n ax m ih s t 0 0 33 410 17 21 2010-10-19 21:34:31.000 ee3bf991-3df0-48ca-94c1-1da84610dbb9
chairman %ch eh r m __% XRMN ch eh r m ax n 0 0 30 30 30 30 2010-10-19 19:18:30.000 f13d0c8d-5dcd-487a-aaa7-905bf70b2dab
president %p r eh z __ d __ n% PRST p r eh z ax d ax n t 0 0 391 391 4 347 2010-10-13 18:03:57.000 f17babdc-7221-46ee-a0bd-b9abf4637033
president %p r eh z __ d __% PRST p r eh z ax d ax n t 0 1 363 391 4 347 2010-10-29 00:13:40.000 f686d7fd-2ae1-41ba-a901-bc19a975393f
inflation %__ n f l ey sh __% ANFL ih n f l ey sh ax n 0 0 14 44 14 14 2010-10-21 00:25:49.000 fb066414-f260-4097-975b-4630f25ccc3f
president bush %p r eh z __ d __ n t b uh% PRST p r eh z ax d ax n t b uh sh 0 0 26 391 0 28 2010-10-19 21:19:41.000 fbbbf62d-fd68-422e-badb-091f74be7bce
presidents clinton and bush %p r eh z __ d __ n t s k l __ n t __ n ae n d b uh% PRST p r eh z ax d ax n t s k l ih n t ax n ae n d b uh sh 0 0 0 391 0 2 2010-10-19 21:15:16.000 ffe2c901-9a62-489b-b477-4deb180d5257

LIST OF REFERENCES

Aho, V. A., & Corasick, M. J. (1975). Efficient string matching: an aid to bibliographic search. *Communications. ACM, 18*(6), 333-340.

Alberti, C., Bacchiani, M., Bezman, A., Chelba, C., Drofa, A., Liao, H., et al. (2009). *An audio indexing system for election video material*. Paper presented at the Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing.

Aleksic, P. S., & Katsaggelos, A. K. (2004). Speech-to-video synthesis using MPEG-4 compliant visual features. *Circuits and Systems for Video Technology, IEEE Transactions on, 14*(5), 682-692.

Arnon, A., Alon, E., & Savitha, S. (2001). *Advances in phonetic word spotting.* Paper presented at the Conference on Information and knowledge management, Atlanta, Georgia, USA.

Ayres, T., & Nolan, B. (2006). Voice activated command and control with speech recognition over WiFi. *Science of Computer Programming*, 109-126.

Berry, T., & Ravindran, S. (1999). *A fast string matching algorithm and experimental results.* Paper presented at the Proceedings of the Prague Stringology Club Workshop`99, Prague.

Bianchi, D., & Poggi, A. (2004). *Ontology based automatic speech recognition and generation for human-agent interaction.* Paper presented at the 13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises

Black, P. E. (2007, 17 December 2007). Soundex. *Dictionary of Algorithims and Data Structures*, from http://xw2k.nist.gov/dads/HTML/soundex.html

Boyer, R. S., & Moore, J. S. (1977). A fast string searching algorithm. *Communications of the ACM*, 337-377.

Carbonell, J., & Goldstein, J. (1998). *The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries.* Paper presented at the Proceedings of the 21st Annual international ACM SIGIR Conference on Research and Development in information Retrieval, Melbourne, Australia.

Cardillo, P. S., Clements, M., & Miller, M. S. (2002). Phonetic Searching vs. LVCSR: How to Find What You Really Want in Audio Archives. *International Journal of Speech Technology, 5*(1), 9-22.

Cardillo, P. S., Clements, M., & Miller, M. S. (2010). USA Patent No. US 7,769,587 B2. G. T. R. Corporation.

Cepstral. (2010). *Cepstral LLC* Retrieved October 20, 2010, from http://www.cepstral.com/

Chelba, C., Silva, J., & Acero, A. (2007). Soft indexing of speech content for search in spoken documents. *Computer Speech and Language, 21*(3), 458-478.

CMU. (2008). Sphinx4 Alpha (Version 4): Carnegie Mellon University.

DeMara, R. F., Gonzalez, A. J., Hung, V., Leon-Barth, C., Dookoo, R. A., Jones, S., et al. (2008). *Towards interactive training with an avatar-based human-computer interface*. Paper presented at the The Interservice/Industry Training, Simulation & Education Conference.

Double Metaphone. (2010, September 21). *Wikipedia, The Free Encyclopedia.* Retrieved October 24, 2010, from http://en.wikipedia.org/w/index.php?title=Double_Metaphone&oldid=386073497

Download Oracle VM VirtualBox (Version 3.2.10). (2010). Oracle.

Fiscus, J. G., Ajot, J., Garofolo, J. S., & Doddington, G. (2006). *Results of the 2006 Spoken Term Detection Evaluation*: NIST. (NIST o. Document Number)

Garofolo, J., Auzanne, G., & Voorhees, E. (2000). *The TREC spoken document retrieval track: A success story.* Paper presented at the Proceedings of the Ninth Text Retrieval Conference (TREC-9).

Gilbert, M., Wilpon, J. G., Stern, B., & Di Fabbrizio, G. (2005). Intelligent virtual agents for contact center automation. *Signal Processing Magazine, IEEE, 22*(5), 32-41.

GoldWave v5.58 Released (Version 5.58). (2010). [Electronic]. St. John's: Gold Wave Inc.

Gurevych, I., Malaka, R., & Porzel, R., & Zorn, H. (2003). *Semantic coherence scoring using an ontology.* Paper presented at the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, Morristown.

Hanna, M. W. (2006). *Topic modeling: beyond bag-of-words*. Paper presented at the Proceedings of the 23rd international conference on Machine learning.

Homonym. (2010). Retrieved October 30, 2010, from http://en.wikipedia.org/w/index.php?title=Homonym&oldid=389601790

Hsinchun, C., & Fei-Yue, W. (2005). Guest Editors' Introduction: Artificial Intelligence for Homeland Security. *Intelligent Systems, IEEE, 20*(5), 12-16.

Huang, H., Feng, C., Wang, J., & Zhang, X. (2010, August). *ASR Normalization for Machine Translation.* Paper presented at the 2nd International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC).

Hung, V., Elvir, M., Gonzalez, A., & DeMara, R. (2009). *Towards a method for evaluating naturalness in conversational dialog systems*. Paper presented at the Proceedings of the 2009 IEEE international conference on Systems, Man and Cybernetics.

Information Retrieval. (2010, October 7). *Wikipedia, The Free Encyclopedia.* Retrieved October 7, 2010, from http://en.wikipedia.org/w/index.php?title=Information_retrieval&oldid=388985341

Jaehui, P., Tomohiro, F., Ikki, O., Hideaki, T., & Sang-goo, L. (2008). *Web content summarization using social bookmarks: a new approach for social summarization*. Paper presented at the Proceeding of the 10th ACM workshop on Web information and data management.

Johnson, S. E., Jourlin, P., G.L, M., Jones, K. S., & Woodland, P. C. (1998). *Spoken Document Retreival for TREC-7 at Cambridge University.* Paper presented at the NIST Special Publication 500-242: The Seventh Text Retrieval Conference (TREC), Washington, DC.

Johnson, S. E., Jourlinz, P., Jonesz, K. S., & Woodlandy, P. C. (1999). *Spoken Document Retreival for TREC-8 at Cambridge Univerisity.* Paper presented at the NIST Special Publication 500-246, Gaithersburg.

Jonathan, M., Bhuvana, R., & Olivier, S. (2007). *Vocabulary independent spoken term detection*. Paper presented at the Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval.

Justin, Z., & Philip, D. (1996). *Phonetic string matching: lessons from information retrieval*. Paper presented at the Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval.

Lawrence, P. (2000). The double metaphone search algorithm. *C/C++ Users J., 18*(6), 38-43.

Le Bigot, L., Terrier, P., Amiel, V., Poulain, G., Jamet, E., & Rouet, J. (2007). Effect of modality on collaboration with a dialogue system. *International Journal of Human-Computer Studies*, 983-991.

Liang, G., Yonggang, D., Wei, Z., & Yuqing, G. (2006). *Integrating Text and Phonetic Information for Robust Statistical Speech Translation.* Paper presented at the Spoken Language Technology Workshop, 2006. IEEE.

Ling, S., Yaohua, G., Yun, W., & Yong, Z. (2009). *Fast Audio Fingerprint Search Strategy for Song Identification*. Paper presented at the Proceedings of the 2009 International Conference on Networking and Digital Society - Volume 02.

Makhoul, J., Kubala, F., Leek, T., Daben, L., Long, N., Schwartz, R., et al. (2000). Speech and language technologies for audio indexing and retrieval. *Proceedings of the IEEE, 88*(8), 1338-1353.

Microsoft. (2009). Microsoft Speech API (SAPI) 5.3: Microsoft Inc.

Microsoft Speech API (SAPI) 5.3 [Electronic. (2009). Version]. *Microsoft MSDN*. Retrieved Dec 22, from http://msdn.microsoft.com/en-us/library/ms723627(VS.85).aspx

NIST. (2008). 2006 Spoken Detection Evaluation. Retrieved March 4, 2010, from http://www.itl.nist.gov/iad/mig//tests/std/2006/index.html

NIST. (2010). NIST Information Access Division, *Evaluation Tools*.

Obermaisser, R., Nah, Y., Puschner, P., Rammig, F., Kim, J. H., Kang, U. G., et al. (2007). Speech Recognition System Using DHMMs Based on Ubiquitous Environment. In *Software Technologies for Embedded and Ubiquitous Systems* (Vol. 4761, pp. 213-222): Springer Berlin / Heidelberg.

OpenCalais. (2009, November 12th 2009). API Metadata - English | OpenCalais. Retrieved November 12, 2009, from http://opencalais.com/documentation/calais-web-service-api/api-metadata

Patman, F., & Shaefer, L. (2003). Is Soundex Good Enough for You? On the Hidden Risks of Soundex-Based Name Searching [Electronic Version], from http://www.immagic.com/eLibrary/ARCHIVES/GENERAL/LAS_US/L030206B.pdf

Phoneme. (2010, October 14). *Wikipedia, The Free Encyclopedia.* Retrieved October 24, 2010, from http://en.wikipedia.org/w/index.php?title=Phoneme&oldid=388243811

Preisach, C., Burkhardt, H., Schmidt-Thieme, L., Decker, R., Schierle, M., Schulz, S., et al. (2008). From Spelling Correction to Text Cleaning – Using Context Information. In H. H. Bock, W. Gaul, M. Vichi, P. Arabie, D. Baier, F. Critchley, R. Decker, E. Diday, M. Greenacre, C. Lauro, J. Meulman, P. Monari, S. Nishisato, N. Ohsumi, O. Opitz, G.

Ritter, M. Schader & C. Weihs (Eds.), *Data Analysis, Machine Learning and Applications* (pp. 397-404): Springer Berlin Heidelberg.

Ramabhadran, B., Sethy, A., Mamou, J., Kingsbury, B., & Chaudhari, U. (2009). *Fast decoding for open vocabulary spoken term detection*. Paper presented at the Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers.

Reddy, D. R. (1976). Speech recognition by machine: A review. *Proceedings of the IEEE, 64*(4), 501-531.

Sang-Hwa, C., Moldovan, D., & DeMara, R. (1993). A Parallel Computational Model for Integrated Speech and Natural Language Understanding. *IEEE Transactions on Computers 42*(10), 1171-1183.

Saraclar, M., & Sproat, R. (2004). Lattice-based search for spoken utterance retrieval. *Proceedings of the HLT-NAACL*, 129-136.

Seide, F., Peng, Y., Chengyuan, M., & Chang, E. (2004). *Vocabulary-independent search in spontaneous speech.* Paper presented at the Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04). IEEE International Conference on.

Seltzer, M. L., Acero, A., & Kalgaonkar, K. *Acoustic model adaptation via Linear Spline Interpolation for robust speech recognition.* Paper presented at the Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on.

Shriberg, E., & Cetin, O. (2006). *Speaker Overlaps and ASR Errors in Meetings: Effects Before, During, and After the Overlap.* Paper presented at the Proceedings of IEEE International

Conference on Acoustics, Speech, and Signal Processing (ICASSP 2006), Toulouse, France.

Singhal, A. (2001). Modern Information Retrieval: A Brief Overview. *IEEE Data Engineering Bulletin 24, 4*, 35-43.

Subramaniam, L. V., Faruquie, T. A., Ikbal, S., Godbole, S., & Mohania, M. K. (2009). *Business Intelligence from Voice of Customer.* Paper presented at the Data Engineering, 2009. ICDE '09. IEEE 25th International Conference on.

Sunday, D. M. (1990). A very fast substring search algorithm. *Communications ACM, 33*(8), 132-142.

Thathoo, R., Virmani, A., Lakshmi, S. S., Balakrishnan, N., & Sekar, K. (2006). TVSBS: A fast exact pattern matching algorithm for biological sequences. *Current Science*, 47-53.

Torkkola, K. (1988). *Automatic alignment of speech with phonetic transcriptions in real time.* Paper presented at the Acoustics, Speech, and Signal Processing, 1988. ICASSP-88.

Wallace, R. G., Vogt, R. J., & Sridharan, S. (2007). *A Phonetic Search Approach to the 2006 NIST Spoken Term Detection Evaluation.* Paper presented at the Proceedings Interspeech 2007 : 8th Annual Conference of the International Speech Communication Association, Antwerp, Belgium.

Wang, Y., Acero, A., & Chelba, C. (2003). Is Word Error Rate a Good Indicator for Spoken Language Understanding Accuracy. *IEEE Xplore,* 577-582.

Weick, K. E. (1976). Educational Organizations as Loosely Coupled Systems. *Administrative Science Quarterly, 21*(1), 1-19.

Wollmer, M., Eyben, F., Keshet, J., Graves, A., Schuller, B., & Rigoll, G. (2009). *Robust discriminative keyword spotting for emotionally colored spontaneous speech using bidirectional LSTM networks.* Paper presented at the Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on.

Woodland, P. C., Johnson, S. E., Jourlin, P., & K. Spärck, J. (2000). *Effects of out of vocabulary words in spoken document retrieval (poster session)*. Paper presented at the Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval.

Yahoo! Developer Network - Developers Resources. (2009, November 12th 2009). Retrieved November 12, 2009, from http://developer.yahoo.com/everything.html

Ying, L., & Yibin, H. (2004). Search audio data with the wavelet pyramidal algorithm. *Inf. Process. Lett., 91*(1), 49-55.

Zechner, K., & Waibel, A. (2000). *Minimizing word error rate in textual summaries of spoken language.* Paper presented at the Proceedings of the 1st North American Chapter of the Association For Computational Linguistics Conference Seattle, Washington.

Zweig, G., Siohan, O., Saon, G., Ramabhadran, B., Povey, D., Mangu, L., et al. (2006). *Automated Quality Monitoring in the Call Center with ASR and Maximum Entropy.* Paper presented at the IEEE International Conference on Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. .