Protecting Privacy in Recorded Conversations

Scot Cunningham
Convergys Corporation
201 E 4th St
Cincinnati, OH 45202
001-513-784-2155
scot.cunningham@convergys.com

Traian Marius Truta
Department of Computer Science
Northern Kentucky University
Highland Heights, KY 41099, USA
001-859-572-7551
trutat1@nku.edu

ABSTRACT
Professionals in the field of speech technology are often constrained by a lack of speech corpora that are important to their research and development activities. These corpora exist within the archives of various businesses and institutions; however, these entities are often prevented from sharing their data due to privacy rules and regulations. Efforts to "scrub" this data to make it shareable can result in data that has been either inadequately protected or data that has been rendered virtually unusable due to the loss resulting from suppression. This work attempts to address these issues by developing a scientific workflow that combines proven techniques in data privacy with controlled audio distortion resulting in corpora that have been adequately protected with minimal information loss.

1. INTRODUCTION
"This call may be recorded for quality assurance purposes." We hear this phrase so often that we have come to ignore it. Few of us think of the implications of this statement. Whenever we call a customer service hotline, our entire conversation may be recorded and retained in a format that is easily accessible to a growing number of individuals.

What exactly is done with these recordings? We assume that "for quality assurance purposes" means that call center supervisors are simply reviewing these recordings in order to ensure that their service representatives (be they human or automated) are interacting properly with their customers. However, our conversations are often used for much more than that.

As speech technology continues to advance, call center recordings are becoming a valuable asset to the companies that possess them. These recordings provide important information to the developers of automated systems about how callers interact with their service representatives and automated systems. Once processed, these recordings have a number of different internal uses.

Potential uses of these recordings include: Natural Language Understanding (NLU) Classifier Model Training [13], Speech Recognizer Model Training [10], Emotion Detection Model Training [6], Construction of Intelligent Agents [28], Speech Application Testing, Automated Feedback Loops & Machine Learning [28].

When combined with various amounts of metadata (e.g. transcriptions and labels), an archive such as this makes a valuable corpus that can be used to help better understand and improve these systems. One popular use of these corpora is as a set of data to train automated speech recognition systems (i.e. decoders – the terms decode and recognize are used interchangeably in this paper). Statistical language models, acoustic models, as well as other probabilistic models all require large amounts of training data for their construction. Speech corpora can also serve as excellent sources of data for use in the research and development of speech technology.

In an effort to improve their systems, owners of corpora often desire to share data with researchers and vendors who can be of great assistance in these efforts. However, corpora such as these usually contain large amounts of sensitive data that should not be shared with such entities. Efforts to scrub this data often incur significant amounts of data loss that can result in corpora that are of diminished value because of information that has been lost in that scrubbing. The lost information includes data values in the corpus transcriptions as well as loss of audio signal characteristics or prosody. Prosodic features such as the pitch and energy in a speaker's voice are valuable information for inferring information about the speaker's emotion and context.

The need to minimize the risk of disclosure and also minimize information loss presents an obvious conflict that is hindering the advance of speech technology in both industry and academia. In an effort to make it possible to share more data, what is proposed here is to utilize the extensive research in data privacy on structured data (see section 1.1, Background and Related Work) and apply that to relatively unstructured speech corpora in order to minimize the disclosure risk of the recorded individuals when sharing that data. The final outcome is a scientific workflow [18] that can enable maximum data sharing and at the same time minimizes the disclosure risk of the individuals represented in that data.

1.1 Background and Related Work

1.1.1 Data Privacy and k-Anonymity
The idea of *k*-anonymity is a concept applied traditionally to structured data. A table satisfies *k*-anonymity if each sequence of quasi-identifiers (the set of attributes that do not directly identify an individual, but used in conjunction with other data sources may lead to disclosure of an individual) appears with at least *k* occurrences in the table [27, 30]. This makes each record indistinguishable from at least *k* - 1 other records. If the data does not meet this *k*-anonymous requirement, then it is at risk for re-identification through quasi-identifiers. For example, Table 1 shows a list of raw data that has not been *k*-anonymized. This is referred to as raw microdata. To make this table *k*-anonymous, it is necessary to generalize the quasi-identifiers so that each sequence of the quasi-identifiers appears with at least *k*

occurrences. Each matching group of *k* occurrences is referred to as an equivalence class.

Table 1 - Raw Microdata

Row	Name	Age	Sex	Zipcode	Disease
1	Scot	5	M	12000	heart disease
2	Marius	9	M	14000	flu
3	Dan	6	M	18000	heart disease
4	Kevin	8	M	19000	flu
5	Richard	12	M	22000	pneumonia
6	Wayne	19	M	24000	pneumonia
7	Kate	21	F	58000	flu
8	Jay	26	F	36000	gastritis
9	Jeff	28	F	37000	pneumonia
10	Liz	56	F	33000	flu

Table 2 - *k*-Anonymized Data

Row	Age	Sex	Zipcode	Disease
1	[1,10]	M	[10001,15000]	heart disease
2	[1,10]	M	[10001,15000]	flu
3	[1,10]	M	[15001,20000]	heart disease
4	[1,10]	M	[15001,20000]	flu
5	[11,20]	M	[20001,25000]	pneumonia
6	[11,20]	M	[20001,25000]	pneumonia
7	[21,60]	F	[30000,60000]	flu
8	[21,60]	F	[30000,60000]	gastritis
9	[21,60]	F	[30000,60000]	pneumonia
10	[21,60]	F	[30000,60000]	flu

Table 2 shows an example of data that has been 2-anonymized and the explicit identifiers (in this case, *Name*) have been removed. Notice that the sensitive data (in this case, *Disease*) has not been altered. Only the quasi-identifiers have been generalized. This protects the privacy of each individual in the table and yet preserves data that can be used for various forms of data mining.

In this work, the concepts of *k*-anonymity are applied to recorded speech, which is much less structured than the tabular data normally associated with *k*-anonymity applications. The same extensions and concerns that apply to *k*-anonymity with structured data also can apply with unstructured audio data. Anyone wishing to apply the concepts of *k*-anonymity to recorded speech should be familiar with these concepts and take the appropriate precautions. These extensions and concerns include: attacks against *k*-anonymity [30, 19], *l*-diversity [19], *p*-sensitive *k*-anonymity [32], *t*-closeness [17], (α, *k*)-anonymity [34], *k*-anonymity with unstructured data [23], multirelational *k*-anonymity [8], etc.

1.1.2 Speech Technology

This paper is related to three distinct areas of speech technology:

1. Automated Speech Recognition (ASR)
2. Natural Language Understanding (NLU)
3. Emotion Detection

ASR and NLU are both important elements for this work. They are necessary for transcribing and classifying audio recordings. Emotion Detection is one possible consumer of this work. These three areas of speech technology are briefly discussed in the following sections.

1.1.2.1 Automated Speech Recognition

Automated speech recognition enables the transcription of recorded telephone conversations. These transcriptions are the raw microdata in this application of data privacy. There are a variety of commercial and open-source toolkits available for automated speech recognition. Several major universities focus entire programs on the research and development of these tools [10, 11, 3] and this work has quickly found its way into commercial development by such notable firms as Microsoft and Nuance. This work is heavily utilized (but not extended) in this paper. For this work, the Sphinx recognizer from Carnegie Mellon University is used as a way to measure the success of distortion techniques.

1.1.2.2 Natural Language Understanding

Natural language understanding focuses on the ability of an automated system to not only recognize the words that are said, but also the meaning of what was said. The CU Communicator Corpus that was obtained for this work [33] was developed with the use of an NLU system to classify each utterance in the corpus. NLU is an area of research that has been ongoing for decades. This technology applies semantic classification trees to obtain one or more classifications for a speech utterance [13]. These classifications are used by speech applications to help understand the meaning of what a user says in order to help guide the interactive dialog between the speech application and the end user. In this paper, these classifications are used to identify quasi-identifiers that need to be generalized or distorted.

1.1.2.3 Emotion Detection

One of the motivators behind this work is the ability to preserve prosodic information for use in emotion detection systems. Emotion detection systems employ a number of different techniques including analyzing words, facial expressions, and the augmented prosodic domain [6]. The prosodic features of most interest to emotion detection are pitch and energy. If these prosodic features can be preserved, then it may be possible to distort the audio and at the same time preserve some of the features necessary for training emotion detection systems. Measuring pitch and energy are important signal processing concepts in this work.

1.1.3 Audio Distortion

This paper applies an experimental distortion technique tailored toward minimizing prosody loss (which represents the information loss for audio files) and maximizing content loss (which lead to disclosure risk minimization). Research in techniques for audio distortion includes several other distortion algorithms which may also be appropriate for this work. These include sample permutation, block permutation, frequency inversion, and a combination of block permutation and frequency inversion [12]. Sample and block permutation both use uniformly distributed permutations to shuffle samples or blocks of samples in order to scramble an audio segment. Frequency inversion

inverts the sign of every other sample. Integrating algorithms such as these with the workflows developed as part of this paper is a suggested area of further study.

One other distortion algorithm is pitch shifting [2]; however, pitch shifting algorithms are designed to incur a high loss of prosody and preserve content. This opposes the goals of this work and is therefore not appropriate in this case. Pitch shifting may, however, be useful if further study is done in the area of speaker identification.

1.2 Contributions

What is proposed in this paper are two scientific workflows designed to protect the privacy of individuals within speech corpora while enforcing the following rules:

1. Reduce Disclosure Risk under a specific threshold
2. Minimize or Measure Information Loss

The first workflow is designed to apply well-known *k*-anonymity techniques to a speech corpus. This is done with the usual goal of achieving *k*-anonymity property (without any background information, the probability of correct disclosure for an individual will be no more than 1/*k*) while reducing information loss. The second workflow is designed to distort the quasi-identifiers in the recorded audio of a speech corpus. Use of distortion, rather than suppression, is a unique approach designed to preserve the prosodic features in the recorded audio. This workflow is designed to minimize loss of prosody (information loss in this case) and distort the quasi-identifiers in the recorded audio up to a predefined threshold thus reducing the risk of disclosure.

Achievement of these goals will result in enabling those who possess valuable speech corpora to be able to more freely share these corpora with other researchers.

The paper is structured as follows. Section 2 describes our general approaches for both transcription generalization and audio distortion. Section 3 presents both scientific workflows and the performed experiments. Conclusions and future research directions are outlined in Section 4.

2. GENERAL APPROACH

The workflows proposed in this paper assume the existence of a speech corpus that contains the following:

- Audio Recordings
- Metadata – This includes transcriptions of the audio recordings and classification labels that identify the audio recordings that hold quasi-identifiers. The transcriptions and classifications can be obtained in a variety of ways. Many corpora obtain these through an approach of automation with human verification in order to ensure accuracy.

2.1 Corpus Metadata Generalization

A method for achieving *k*-anonymity or a more enhanced privacy model (several models were listed in section 1.1.1) will be used to properly generalize the metadata in the corpus. This is referred to as the generalization algorithm. Various algorithms that achieve *k*-anonymity property (or an enhanced model) while trying to minimize an information loss exist [15, 16, 1, 31, etc.]. This paper does not mandate any one privacy model and a specific algorithm although *k*-anonymity and the greedy clustering algorithm introduced by Byun, Kamra, Bertino, and Li [1] were selected for empirical study.

The metadata in speech corpora come in a variety of forms. The designer of the corpus decides how the metadata will be represented. If the *k*-anonymity method is not able to take the format of the metadata as input, then some conversion of that metadata is obviously necessary.

Once the corpus metadata is prepared for input to the generalization algorithm, then the algorithm is executed to generalize the values of the quasi-identifiers. Some post-processing may be necessary to incorporate the generalized values back into the metadata. The list of quasi-identifiers is also retained for use in audio distortion as shown in the next section.

The generalization algorithm should report the amount of information lost. This provides consumers of the generalized corpus information about the quality of the corpus content. There are various information loss measures proposed in the literature [20, 9, 35, etc.]. While any of them can be used in our workflow, we use the one presented in [1].

2.2 Iterative Distortion

Distortion (random noise [22], etc.) and suppression [30] are two techniques that are used in data privacy research probably to a lesser extent than generalization. However, both of these techniques or a combination of them is a good choice when it comes to the audio portion of a speech corpus. A speech corpus is usually distributed as a combination of audio data and associated metadata that can be linked to the audio files. Because generalized values for quasi-identifiers are retained in the metadata, there is no need to perform the same generalizations in the audio recordings. If the generalized data is needed, one can simply refer to the metadata. The aim in protecting privacy in the audio portion of the corpus is making the quasi-identifiers that appear in the audio difficult to recognize (reducing disclosure risk) while minimizing loss of prosody (information loss). The steps to achieve this are described in the following sections.

2.2.1 Prepare Time-Aligned Word Boundaries

The quasi-identifiers that were generalized in the corpus metadata must be identified in the audio recordings. Within each audio recording, each quasi-identifier must be identified along with its time-aligned word boundary.

A time-aligned word boundary provides the text of the word that was spoken along with time markers that mark the beginning and ending time of the word spoken. These time markers measure the amount of time from the beginning of the recording. Each audio file must have time-aligned word boundary prepared for it. This provides the information needed to distort each generalized quasi-identifier. Figure 1 illustrates a time-aligned word boundary.

Start	End	Text
1.6315944	2.1368176	Raleigh
6.6751760	7.4312807	Chicago

Figure 1 – Example Time-Aligned Word Boundary

2.2.2 Distort Quasi-Identifiers

Distortion of the quasi-identifiers must be done in such a way as to 1) thwart the ability of a recognizer to identify the spoken words, and yet 2) still preserve as much prosodic information as possible. Although popular speech encryption techniques satisfy the first goal, they are not designed to preserve prosody. This paper describes a new distortion technique that is designed to preserve prosody.

De-identifying the audio signal focuses on a combination of *controlled randomization* of digital samples combined with iterative amounts of *suppression* necessary to thwart the use of an automated speech recognizer. In the first step, controlled randomization, the samples are randomized within an acceptable range. To illustrate, assume that Figure 2 is the waveform of a signal that to distort.

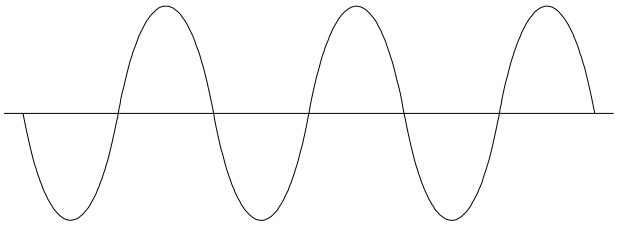

Figure 2 - Simple Sine Wave Tone

Of course this is a simple sine wave. A typical speech signal is a much more complex waveform. However, this sine wave is useful for illustrating this distortion technique and the same technique can be used with more complex waveforms. As discussed earlier, a waveform is represented in the computer by a series of samples as illustrated in Figure 3.

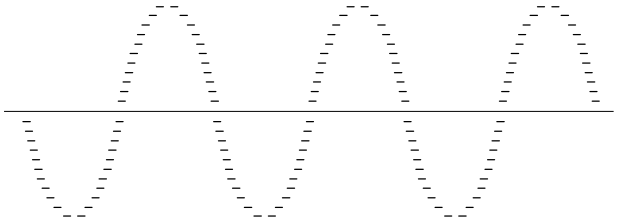

Figure 3 - Digital Samples of Sine Wave

Controlled randomization begins by defining an acceptable range within which to allow randomization of the signal samples. For example, the solid curve in the Figure 4 illustrates this range.

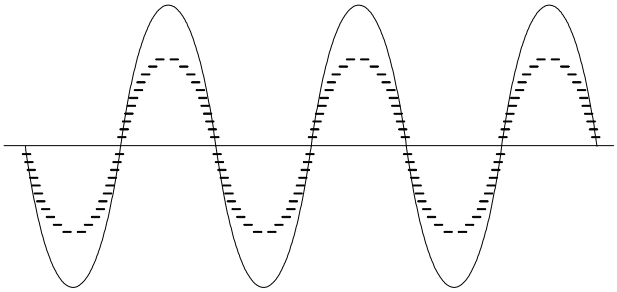

Figure 4 - Defining the randomization range

Next, each sample is randomized between zero and the defined range for that sample. For example, if the original sample is -50 and the randomization range is 1.5 times the sample, then the new distorted sample is a random number between 0 and -75. This is illustrated in Figure 5.

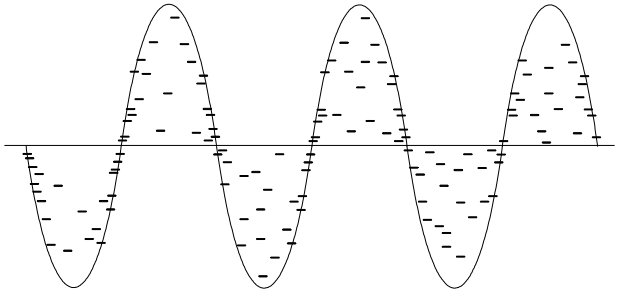

Figure 5 - Assigning random values to each sample

This results in a set of randomized samples that preserve the general shape of the waveform but yet add a good deal of noise and distortion to the audio signal. If the curve that defines the acceptable range is removed, the distortion, as well as the preservation of some signal characteristics, becomes more obvious as shown in Figure 6.

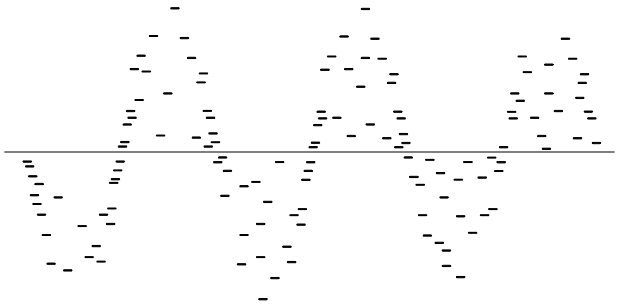

Figure 6 - Randomized Samples

Unfortunately, this controlled distortion technique preserves a good deal of the fundamental audio characteristics necessary to distinguish speech content. This is mainly due to the fact that samples that are closer to the silence axis have a smaller range for movement when they are randomized, thus preserving much of the lower-level signal content. In order to eliminate this problem, the lower level signals are suppressed (Figure 7). This is referred to as *suppression*.

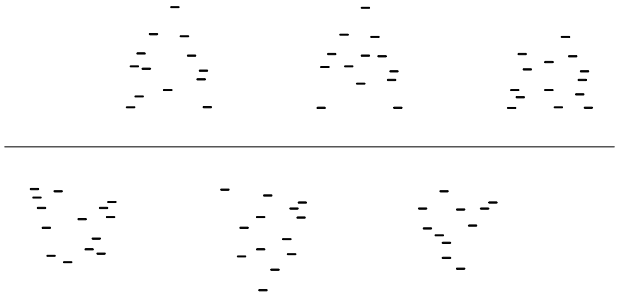

Figure 7 - Lower Level Samples Suppressed

The use of controlled randomization plus suppression results in a set of randomized samples that preserves some basic signal characteristics of the original audio, but yet is distorted enough to obscure any recognizable speech and thus minimize disclosure risk in the actual audio recording. The trick here is to define the optimal *silence range* around the axis to suppress. When a silence range has been selected, the function *d(a,s)* will distort the audio segment *a* using a silence range of *s* using the techniques described so far in this section.

Based on experiments performed, there appears to be no single silence range that is suitable to obscure all varieties of speech. The proper silence range is highly dependent on the pitch and energy of the speaker's voice at the time of the utterance that is being distorted. This is because varying frequencies and amplitudes (in the same speaker as well as between speakers) can increase or decrease the slope of the audio signal around the silence axis. This has a direct affect on the size of the range made available for randomization in this algorithm. Therefore, an iterative approach is necessary to identify the optimal silence range necessary to thwart the ability of a speech recognizer to identify the word(s) being spoken. This iterative distortion algorithm is defined in Figure 8.

In the distortion algorithm, a speech recognizer is used to generate a hypothesis for the utterance that has been distorted using *w(a)* (line 10). The hypothesis is scored by comparing it with the transcription of the original audio in order to obtain the word error rate (WER) [21]. Since in our algorithm there are no inserted or deleted words, the WER for an utterance is the number of incorrectly identified words over the total number of words.

For example, consider the following two transcriptions:

Original: "I WANNA GO TO CHICAGO ON DECEMBER FIFTEENTH LEAVING IN THE MORNING"
Hypothesis: "I WANNA GO TO ******* ** THE FIFTEENTH LEAVING IN THE MORNING"

The original phrase contains 12 words. The hypothesis from the recognizer was incorrect on 3 of those words. (Distortion can sometimes disturb surrounding words due to inaccurate word boundaries and the nuances of statistical language models.) It had no hypothesis for the words CHICAGO and ON (indicated by asterisks) and an incorrect hypothesis THE for the word DECEMBER. The WER in this case is 3/12, or 25%.

Once the WER of the distorted audio is greater than the WER of the original audio, the audio has been altered enough to cause confusion in the recognizer to the point that it can no longer understand the phrase that was distorted. That completes the iteration at which point the required level of distortion bas been achieved. The distorted audio is retained and total prosody loss is output (line 18).

The accuracy of the speech recognizer must be optimized in order to obtain good recognition results. By training the speech recognizer on the entire set of utterances to be distorted, a heightened level of accuracy can be achieved. Normally, recognizers are trained on a training set that is assembled to best represent what the recognizer should expect during decoding. However, in this case, it is known in advance *exactly* what the recognizer will need to understand. (This assumes of course that complete transcriptions of the utterances are available.)

Definitions:

d(a,s): distort audio *a* with silence range of *s* and return the distorted utterance

s(a): suppress audio *a* (i.e. replace with complete silence)

w(a): decode (i.e. recognize) the audio *a* and return the word error rate (WER)

p_rmsd(a,b): Compute the Root Mean Square Deviation (RMSD) between the pitch of a and the pitch of b

e_rmsd(a,b): Compute the RMSD between the energy of a and the energy of b

Distortion Algorithm:

1. Train the speech recognizer on the entire set of utterances to be distorted
2. For each utterance a_u to be distorted
3. $s = 0$ // Initial silence range of 0
4. $e_d = 0$ // Cumulative rmsd for distorted energy
5. $p_d = 0$ // Cumulative rmsd for distorted pitch
6. $e_s = 0$ // Cumulative rmsd for suppressed energy
7. $p_s = 0$ // Cumulative rmsd for suppressed pitch
8. $wer_u = w(a_u)$ // Obtain undistorted word error rate (WER)
9. $a_d = d(a_u,s)$ // Create distorted audio file
10. while $w(a_d) \leq wer_u$
11. $s = s + k$ // Increment silence range by constant *k*
12. $a_d = d(a_u,s)$ // Create distorted audio
13. $a_s = s(a_u)$ // Create suppressed audio
14. $p_d = p_d + p\_rmsd(a_u, a_d)$
15. $p_s = p_s + p\_rmsd(a_s, a_d)$
16. $e_d = e_d + e\_rmsd(a_u, a_d)$
17. $e_s = e_s + e\_rmsd(a_s, a_d)$
18. *Output* p_d, e_d, p_s, e_s // Quantify total prosody loss

Figure 8 - Distortion Algorithm

2.2.3 Minimize Disclosure Risk

Risk of disclosure is minimized by measuring the capability of an automated recognizer to discern what was said in the distorted audio. (Human recognition is an area of further study.) This is done in line 10 of the distortion algorithm presented in Figure 8. Here, the recognition performed on the distorted audio, $w(a_d)$, returns a WER which is compared against the WER from the undistorted audio, wer_u. Because the audio being distorted is limited to only the words identified in the time-aligned word boundary, we know that increases in the WER are a result of the recognizer's inability to discern the targeted audio.

2.2.4 Measure Pitch and Energy

To measure the information lost by the distortion algorithm in terms of prosody, it is necessary to measure the energy and pitch characteristics of the suppressed and distorted audio samples. These measurements are taken in lines 14 through 17 of the algorithm (see Figure 8).

Energy is represented by the amplitude of the speech signal and is perceived as the volume in decibels. Since each sample has an individual amplitude, energy of an interval of speech can be measured with the root mean square (RMS) of all the samples within that interval.

Pitch contour is measured as a vector of values where each value is the average pitch for a speech segment. The ability to measure changes in pitch is somewhat complex, but measured over time

can add value in understanding more about what is being said. For example, an upward direction in pitch at the end of an English sentence usually indicates that a question is being asked. Changes in pitch are also very useful in detecting the emotions being expressed by the speaker. The acoustic characteristic that is most closely associated with pitch is the fundamental frequency of the sound wave which, in speech, is determined by the frequency of vibration of the vocal cords [7]. There are many techniques for estimating the pitch of a speech segment given a waveform [25]. One popular method to estimate the pitch of a speech segment of size N is the average magnitude difference function (AMDF) [26] and is defined as:

$$AMDF(\tau) = \frac{1}{N}\sum_{n=0}^{N-1} | s(n) - s(n-\tau) | \quad (1)$$

where is close to the size of the fundamental period and s(n) is the value of the nth sample within segment N. In speech, the fundamental period can be defined as the time between the opening and closing of the glottis. There are many algorithms for estimating the fundamental period that go beyond the scope of this work.

2.2.5 Measure Prosody Loss

While generalizing the corpus metadata, information loss is measured in terms of the amount of generalization that takes place. For the distorted audio, the concern is the preservation of prosodic information or the measurement of prosody loss. This is accomplished by comparing vectors of prosodic information from the original audio with prosodic vectors of the distorted audio. A Root Mean Square Deviation algorithm (see equation 2) is employed to calculate the distance between these vectors.

$$RMSD(a,b) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(a_i - b_i)^2} \quad (2)$$

This formula has two instantiations, for pitch contour and energy (labeled *p_rmsd* and *e_rmsd* in Figure 8).

3. EMPIRICAL STUDY

Experiments were conducted on a PC configured with a Linux 2.6.19 kernel and 768 MB of memory, and 40 GB of disk space. The corpus used is the CU Communicator Corpus [33] which contains recordings and metadata for an automated speech travel reservation system.

The experiments were broken into two phases:

1. Corpus Metadata Generalization
2. Audio Distortion

The details of these experiments are described next.

3.1 Corpus Metadata Generalization

A Byun Clustering software implementation described in [1] was used for these experiments. The first challenge was to take the data provided by the CU Communicator corpus and transform it into a format usable as input by the Byun Clustering software. The software requires a simple text input file of tab-delimited data.

The CU Communicator corpus consists of recordings that have been previously transcribed, segmented, and classified. Each recording is an individual segment of an interaction between a caller and the automated CU Communicator system. The CU Communicator system is a travel reservation system that captures, among other things, arrival and departure cities. The transcriptions and classifications are provided in XML format along with a very large amount of other data that is not needed for these experiments.

The challenge here is to convert the data in this XML into a more structured format that can be easily used by the Byun Clustering software as described above. An XSL transformation (XSLT) stylesheet was used to select the data of interest and a relational database was created from that data. This database was populated with 31,790 transcriptions. Arrival and departure cities were selected as the quasi-identifiers.

Natural Language Understanding (NLU) classifications from the corpus were used to identify the quasi-identifiers. To generalize arrival and departure cities, a generalization hierarchy was created with four levels as depicted in Figure 9.

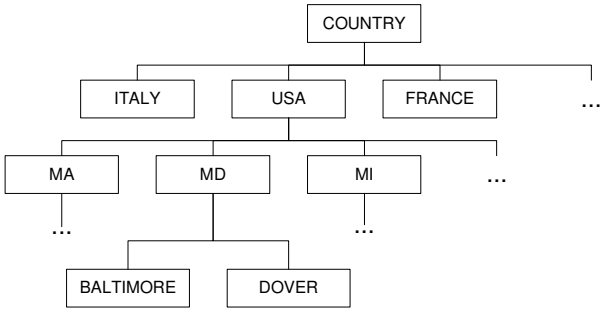

Figure 9 - City Generalization Hierarchy

The Byun Clustering software was then run with *k*=3 to generalize the quasi-identifiers for each transcription. The positions of the generalized quasi-identifiers were located within each audio file and a .lab file was created to record each of those positions. These .lab files were used as input to the audio distortion phase described in the next section. The entire workflow for generalizing the corpus metadata is depicted in Figure 10.

In the case of our experiments, the generalization hierarchy illustrated in Figure 9 is the data source labeled "Generalization Hierarchies" in Figure 10. This hierarchy is used in the Byun Clustering Generalization.

Generalization resulted in an average information loss of 0.25 per quasi-identifier. Information loss during generalization is calculated as follows: If n is the number of levels in the generalization hierarchy and l_{qi} is the level that the QI was generalized to, then information loss for that quasi-identifier is calculated as specified in equation 3.

$$IL_{qi} = l_{qi}/n \mid 0 \le l \le n \quad (3)$$

For example, in Figure 9, if DOVER is generalized to USA, then l_{qi} is 2 (USA) and n is 3 (three possible generalization levels in the hierarchy) resulting in an information loss of 2/3. If DOVER were to remain unchanged, then l_{qi} would be 0 resulting in an information loss of 0.

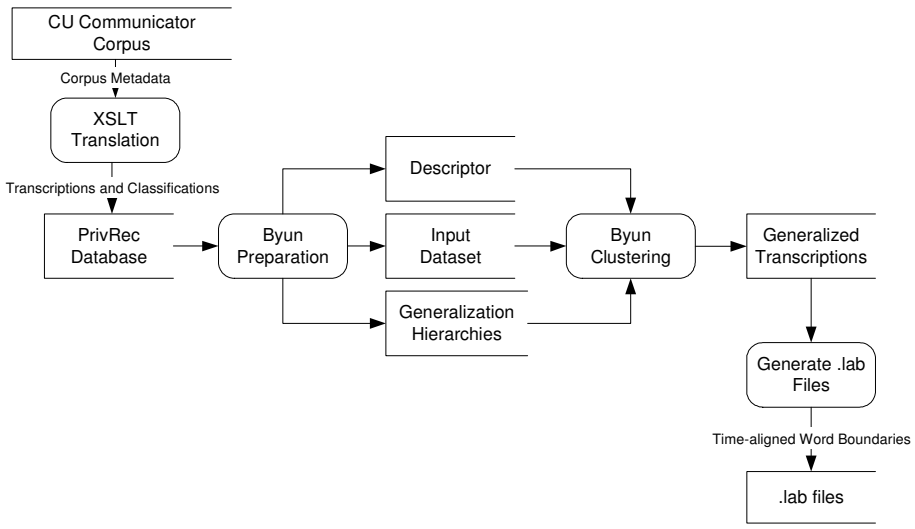

Figure 10 - Generalization Workflow

The total calculations for information loss are based on the information loss for each quasi-identifier. Table 3 illustrates an example of the information loss calculations for all quasi-identifiers where the departure city was "DENVER".

Table 3 - Example Information Loss Calculation

Original Value	Generalized Value	Number of Occurrences	IL Per Occurrence	Total IL
DENVER	DENVER	595	0.00	0.00
DENVER	CO	5	0.33	1.65
DENVER	USA	119	0.66	78.54
DENVER	COUNTRY	45	1.00	45.00
Totals		**764**		**125.19**

Avg.IL/QI = 125.19/764 = 0.16

For this subset of data, the average information loss (IL) per quasi-identifier (QI) is 0.16. Since the number of occurrences of DENVER as a departure city in this corpus is quite high, it makes sense that less generalization was required for this subset than for the overall corpus because less generalization would be required in order to achieve *k*-anonymity. A complete description of information loss measure can be found in [1, 31].

The generalization performed thus far has been applied only to the transcriptions accompanying the recorded audio in the corpus. Our next step is to apply a similar level of protection to the associated recorded data through distortion.

3.2 Audio Distortion

Audio file distortion was accomplished using custom-developed software integrated with well-known open-source components including:

- *WaveSurfer 1.8.5* [29]
- *sph2pipe 2.5* [14]
- *MySQL 4.1.20*
- *CMU-Cambridge Statistical Language Modeling Toolkit* v2 [5]
- *Sphinx 3.7.0 Decoder* [3]
- *SphinxTrain 1.0 Training Kit* [3]
- *Sphinx Knowledge Base Tool* [4]
- *NIST Speech Recognition Scoring Toolkit (SCTK) 2.2.4* (sclite scoring tool) [24]
- *get_f0* program used to calculate RMSD for both pitch contour and energy. [29]

The custom software components developed for this workflow included:

- *distort* – a C language program used to implement the distortion technique discussed in section 2.2.2
- *rmsd* – a C language program that implements the RMSD algorithm
- *distort_test.sh* – a UNIX shell script that implements the iterative workflow described in this section
- *comparef0* – a UNIX shell script that utilizes the get_f0 program to compare the prosody and energy vectors of two different audio files

These custom components, as well as links to the referenced 3rd party components, are available as links from the website for this project at http://cscdb.nku.edu/privrec.

The iterative workflow (implemented in *distort_test.sh*) was able to successfully distort the audio so that a recognizer (Sphinx) could no longer perform accurate recognition. In addition, many of the original prosodic features were preserved. This iterative distortion workflow is illustrated in Figure 11.

For these tests, the Sphinx recognizer was trained on the audio and transcriptions that were to be distorted. The detailed steps required to set up the recognizer are described in [3]. Normally, a training set consists of recorded audio and associated transcriptions that are representative of the audio that the recognizer can expect. For this testing, the content of all audio that will be presented to the recognizer is known in advance. This provides the recognizer with highly accurate training data that should result in very accurate recognition rates necessary for this processing.

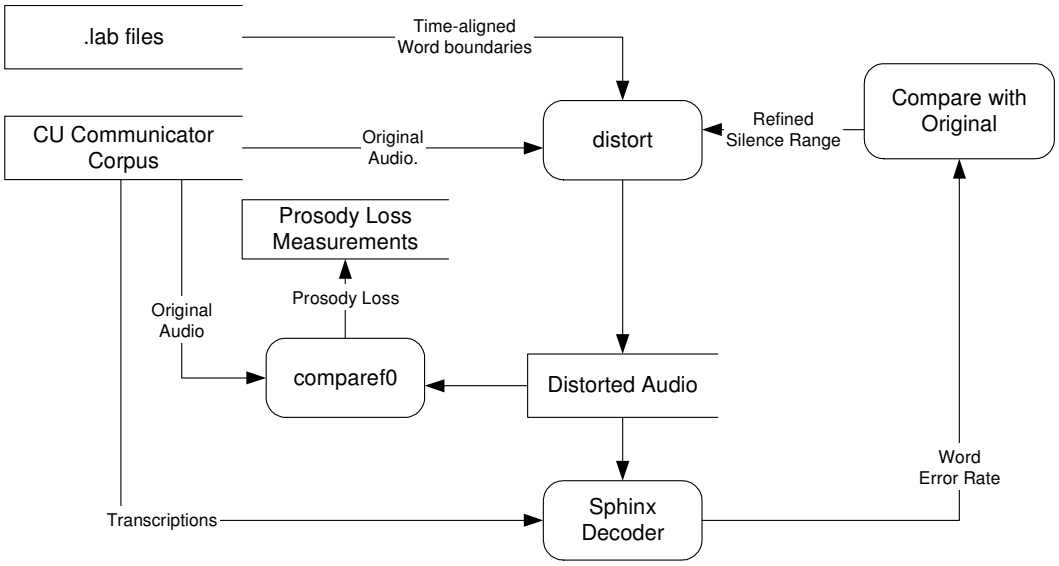

Figure 11 - Iterative Distortion Workflow

The distort program uses time-aligned word boundaries from the .lab files to identify the position of each word to be distorted. The distort program first distorts the original audio using an initial silence range. The distorted audio is then run through the Sphinx Decoder and the word error rate (WER) is calculated. If the WER has not exceeded the WER from original audio, the silence range is increased and the process is repeated.

The distorted audio is also run through the *comparef0* script to measure and compare pitch and energy vectors of the distorted audio with the original audio in order to measure prosody loss. RMSD is used to compare the distorted audio and the original audio and calculate the total prosody loss for both pitch and energy. This is compared with the total prosody loss measurements from suppression in order to determine the amount of improvement gained by the controlled distortion algorithm.

The resulting distorted audio was compared with audio in which the quasi-identifiers had been completely suppressed with silence. When compared with suppression, controlled distortion showed a 47.5% improvement in prosody loss and a 75% improvement in energy loss.

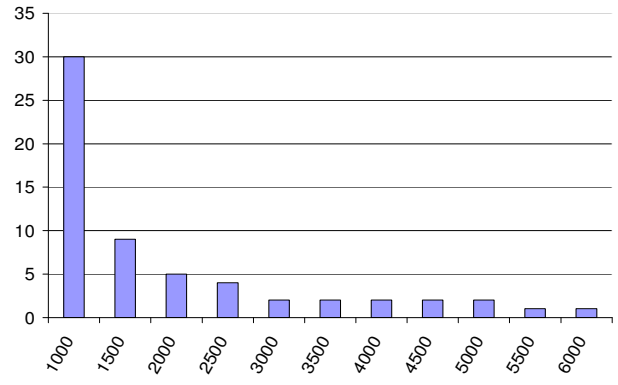

Figure 12 - Silence Ranges

The silence ranges necessary to minimize disclosure risk varied across the test set of utterances (see Figure 12). About half the utterances achieved the necessary amount of distortion with the initial range of 1000. So, for example, a 16-bit sample with a value of 999 would be silenced. A sample with a value of 1000 would be left alone. The other half of the utterances required anywhere from 1500 to 6000 for a silence range.

4. CONCLUSIONS AND FUTURE WORK

For purposes of research and development of speech technology, as well as for the improvement of existing speech systems, it is often necessary to share corpora among various business and academic entities. However, minimizing disclosure risk in these recordings and any associated transcriptions is an important consideration. These conflicting goals often hinder progress in the area of speech technology. Common approaches to this problem include complete suppression of the sensitive portions of the audio. This is a time consuming effort and can result in significant loss of prosody. Preserving prosody can be very useful for purposes such as gender identification, age identification, and emotion detection.

The purpose of this research was to integrate work that has been done in the area of data privacy with speech recognition and distortion to develop scientific workflows for protecting privacy in recorded conversations with minimal loss to data content and prosodic features in the audio. Minimizing disclosure risk is the most important goal with the second being to minimize/measure information and prosody loss. This enables the comparison of different techniques for generalization and distortion.

The CU Communicator Corpus was used for testing. This corpus contained speech recordings that had already been transcribed and classified. The classifications were used to identify transcriptions that contained quasi-identifiers and separate out those quasi-identifiers into tabular fields in a relational database. A data privacy algorithm (Byun Clustering) was run against the quasi-identifiers to generalize them. On average, each quasi-identifier was moved less than one step up

the generalization hierarchy. These generalized quasi-identifiers were then used as input to a process that identified words within the transcriptions that needed to be distorted in order to protect the audio content associated with that transcription.

The distortion workflow was developed that took the output of the generalization phase and proceeded to apply iterative levels of distortion to the quasi-identifiers in the associated audio. The intent of this distortion was to thwart the ability of an automated speech recognition system (in this case, Sphinx) to successfully decode the quasi-identifiers. By iteratively increasing distortion and suppression, the minimal amount of distortion was applied to increase the word error rate (WER) that signified that the recognizer was no longer able to recognize the audio segment. Loss of prosody in the audio was measured using an RMSD algorithm to compare the prosodic vectors of the distorted audio with those of the original audio. This was compared to the loss resulting from completely suppressing quasi-identifiers in the audio. Results indicated improvements of as much as 75% over suppression.

The manner in which this workflow was developed enables the integration of any form of data privacy or audio distortion. This helps open up the door to other avenues of further research. Applying this research to various corpora and data privacy alternatives would be a productive area of study, especially for those who have access to corpora that contain actual sensitive information (most publicly and commercially available corpora do not).

For this work, a custom distortion technique was developed aimed at preserving prosodic features. This distortion technique combined controlled randomization with a user-specified level of suppression of lower-level audio samples. By iteratively increasing the amount of suppression, the workflow found an optimal level of suppression needed to increase the WER. Alternative distortion techniques should be studied to ascertain their effectiveness on preserving prosody and thwarting recognition, be it automated recognition or human intelligibility. As was the case with this work, it is expected that thwarting intelligibility and preserving data will be conflicting goals that will need to be balanced.

High quality automated recognition is important to the ability of these algorithms and data flows to perform well. Further research could be done in this area to better tune automated recognizers and perhaps experiment with several different recognition engines to achieve optimal performance. Iteratively tuning the recognizer with audio that has already been distorted may also improve results.

Using a combination of third party speech recognition systems, various *k*-anonymity generalization algorithms, custom distortion software, and off-the-shelf hardware, experiments were conducted that accurately identified the data that should not be shared, generalized the corpus transcriptions, and distorted the associated audio segments while measuring loss of transcribed data as well as loss of the prosodic features in the audio.

Although there are still some challenges remaining, a scientific workflow was successfully developed for approaching this speech privacy problem. Further research can be performed to refine the presented techniques. The goal is that this will enable more corpora and higher quality corpora to be shared and made readily available to researchers and developers in the field of speech technology while minimizing disclosure risk and information (prosody) loss.

Finally, an area that was briefly mentioned, speaker identification, is worth further study. The speaker's voiceprint can be considered an identifier that needs to be addressed.

Acknowledgments

The work presented herein was completed by Scot Cunningham while he was a graduate student in the Department of Computer Science at Northern Kentucky University (NKU), Highland Heights, USA. The authors would like to thank Kevin Kirby and Richard Fox for their suggestions. The authors would also like to thank the Center for Spoken Language Research (CSLR) at the University of Colorado for making the CU Communicator Corpus [33] available to NKU for this research effort. The CU Communicator Corpus was a valuable asset to this work.

5. REFERENCES

[1] Byun J.W.; Kamra A.; Bertino E.; Li N.: "Efficient k-Anonymization using Clustering Techniques". *Proc. of DASFAA*, 2007, 188–200.

[2] Chaudhari, J.; Cheung, S.; Venkatesh, M.: "Privacy Protection for Life-log Video". *IEEE Workshop on Signal Processing Applications for Public Security and Forensics*, April 2007, 1–5.

[3] CMU: Sphinx Group Open Source Speech Recognition Engines, http://cmusphinx.sourceforge.net/html/cmusphinx.php.

[4] CMU: Sphinx Knowledge Base Tool, http://www.speech.cs.cmu.edu/tools/lmtool.html.

[5] CMU: Statistical Modeling Toolkit, http://www.speech.cs.cmu.edu/SLM_info.html.

[6] Cowie, R.; Douglas-Cowie, E.: Tsapatsoulis, N.; Votsis, G.; Kollias, S.; Fellenz, W.; Taylor, J.G.: "Emotion Recognition in Human-Computer Interaction". *IEEE Signal Processing Magazine*, Vol.18, No.1, 2001, 32–80.

[7] Denes P.B.; Pinson E.N.: "The Speech Chain: The Physics and Biology of Spoken Language". *W.H. Freeman and Company*, ISBN 0-7167-2344-1, 2007.

[8] Ercan Nergiz M.; Clifton C.; Erhan Nergiz A.: "MultiRelational k-Anonymity". *Proc. of IEEE ICDE*, 2007, 1417–1421.

[9] Ghinita G.; Karras K.; Kalinis P.; Mamoulis N.: "Fast Data Anonymization with Low Information Loss". *Proc. of VLDB*, 2007, 758–769.

[10] Hain T.: "Automatic Transcription of Conversational Telephone Speech". *Cambridge University Engineering Department Technical Report*, December, 2003.

[11] HTK: Cambridge University Engineering Department, 2006, http://htk.eng.cam.ac.uk/.

[12] Jayant, N.; McDermott, B.; Christensen, S.; Quinn, A.: "A Comparison of Four Methods for Analog Speech Privacy". *IEEE Transactions on Communications*, Vol. 29, No. 1, 1981, 18 – 23.

[13] Kuhn R.: “The Application of Semantic Classification Trees to Natural Language Understanding”. *IEEE Transaction on Pattern Analysis and Machine*, Vol. 17, No. 5, 1995.

[14] LDC: sph2pipe: ftp://ftp.ldc.upenn.edu/pub/ldc/misc_sw/sph2pipe_v2.5.tar.gz.

[15] LeFevre K.; DeWitt D.; Ramakrishnan R.: “Incognito: Efficient Full-Domain *k*-Anonymity”. *Proc. of the ACM SIGMOD*, Baltimore, Maryland, 2005, 49–60.

[16] LeFevre K.; DeWitt D.; Ramakrishnan R.; Mondrian: “Multidimensional *k*-Anonymity”. *Proc. of the IEEE ICDE*, Atlanta, 2006.

[17] Li N.; Li T.; Venkatasubramanian S.: “t-Closeness: Privacy Beyond k-Anonymity and l-Diversity”. *Proc. of the IEEE ICDE*, 2007, 106–115.

[18] Ludscher B.; Altintas I.; Berkley C.; Higgins D.; Jaeger E.; Jones M.; Lee E.; Tao J.; Zhao Y.: “Scientific Workflow Management and the Kepler System”. *Concurrency and Computation: Practice & Experience*, 2005.

[19] Machanavajjhala A.; Gehrke J.; Kifer D., Venkitasubramaniam M.: “l-Diversity: Privacy Beyond k-Anonymity”. *Proc. of the IEEE ICDE*, Atlanta, 2006, 24.

[20] Mateo-Sanz J.M.; Domingo-Ferrer J.; Sebe F.: “Probabilistic Information Loss Measures in Confidentiality Protection of Continuous Microdata”. *Data Mining and Knowledge Discovery*, Vol. 11, No. 2, 2005, 181–193.

[21] McCowan I.; Moore D.; Dines J.; Gatica-Perez D.; Flynn M.; Wellner P.; Bourlard H.: “On the Use of Information Retrieval Measures for Speech Recognition Evaluation”. *Technical Report IDIAP-RR 04-73*, Martigny, Switzerland, 2004.

[22] Muralidhar K.; Sarathy R.: “Security of Random Data Perturbation Methods”. *ACM Transactions on Database Systems*, Vol. 24, No. 4, 1999, 487–493.

[23] Newton E.; Sweeney L.; Malin B.: “Preserving Privacy by De-identifying Facial Images”. *IEEE Transactions on Knowledge and Data Engineering,* Vol. 37, No. 3, 2005, 179-192.

[24] NIST: NIST Spoken Language Technology Evaluation and Utility, http://www.nist.gov/speech/tools/index.htm.

[25] Rabiner L.R.; Cheng M.J.; Rosenberg A.E.; McGonegal C.A.: “A Comparative Performance Study of Several Pitch Detection Algorithms”. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-24, 1976, 399–418.

[26] Ross M.J.; Shaer H.L.; Cohen A.; Freudberg R.: “Average Magnitude Difference Function Pitch Extractor”. *IEEE transactions on Acoustics, Speech, and Signal Processing*, ASSP-22, 1974, 353–362.

[27] Samarati P.; "Protecting respondents' identities in microdata release". *IEEE Transactions on Knowledge and Data Engineering,* Vol. 13, No. 6, 2001, 1010–1027.

[28] Sahin, F.; Bay, J.S.: "Learning from Experience Using a Decision-Theoretic Intelligent Agent in Multi-Agent Systems". *Proc. of the IEEE Mountain Workshop on Soft Computing in Industrial Applications*, 2001, 109–114.

[29] Sjölander, K.; Beskow, J.: "WaveSurfer - An Open Source Speech Tool". *International Conference on Spoken Language Processing*, Beijing, China, 2000, 464–467.

[30] Sweeney L.: “Achieving k-Anonymity Privacy Protection Using Generalization and Suppression”. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, Vol. 10, No. 5, 2002, 571–588.

[31] Truta T.M.; Campan A.: “K-Anonymization Incremental Maintenance and Optimization Techniques”. *Proc. of the ACM SAC*, 2007, 380–387.

[32] Truta T.M.; Vinay B.: “Privacy Protection: p-Sensitive k-Anonymity Property”. *International Workshop of Privacy Data Management*, 2006.

[33] Ward W.; Pellom B.: "The CU Communicator System," *IEEE Workshop on Automatic Speech Recognition*, Keystone Colorado, 1999.

[34] Wong R.; Li J.; Fu A.; Wang K.: “(α, k)-Anonymity: An Enhanced k-Anonymity Model for Privacy Preserving Data Publishing”. *Proc. of the ACM SIGKDD*, 2006, 754–759.

[35] Xu J.; Wang W.; Pei J.; Wang X.; Shi B.; Fu A.: “Utility-Based Anonymization Using Local Recoding”. *Proc. of ACM SIGKDD*, 2006, 785–790.