Fast Transcription of Unstructured Audio Recordings

Brandon C. Roy, Deb Roy

The Media Laboratory
Massachusetts Institute of Technology
Cambridge, Massachusetts, USA
bcroy@media.mit.edu, dkroy@media.mit.edu

Abstract

We introduce a new method for human-machine collaborative speech transcription that is significantly faster than existing transcription methods. In this approach, automatic audio processing algorithms are used to robustly detect speech in audio recordings and split speech into short, easy to transcribe segments. Sequences of speech segments are loaded into a transcription interface that enables a human transcriber to simply listen and type, obviating the need for manually finding and segmenting speech or explicitly controlling audio playback. As a result, playback stays synchronized to the transcriber's speed of transcription. In evaluations using naturalistic audio recordings made in everyday home situations, the new method is up to 6 times faster than other popular transcription tools while preserving transcription quality.
Index Terms: speech transcription, speech corpora

1. Introduction

Speech transcription tools have been in use for decades, but unfortunately their development has not kept pace with the progress in recording and storage systems. It is easier and cheaper than ever to collect a massive multimedia corpus, but as the size of the dataset grows so does the challenge of producing high quality, comprehensive annotations. Speech transcripts, among other annotations, are critical for navigating and searching many multimedia datasets.

1.1. Speech transcription

Our approach to speech transcription is to leverage the complementary capabilities of human and machine, building a complete system which combines automatic and manual approaches. We call this system *BlitzScribe*, since the purpose of the tool is to enable very rapid orthographic speech transcription.

To situate our system relative to other approaches, we consider transcription along several key dimensions. Functionally, transcription may be automatic or manual. Automatic methods require little human oversight but may be unreliable, while manual methods depend on human labor but may be excessively time consuming or expensive. Another dimension is the granularity of time-alignment between the words and the audio. For example, a transcript of an hour long interview might be a single document with no intermediate time stamps, or each word could be aligned to the corresponding audio. Time-aligned transcripts require significantly more information, and thus more human effort if generated manually. A third aspect is whether the transcription must be performed in real time or offline. For example, courtroom stenographers and closed-captioning (subtitling) of live broadcasts must be performed in real time. Stenographers can typically transcribe 200-300 words per minute [1] using a specialized keyboard interface, but it may take years to develop this proficiency. Recently, a method of "re-speaking" has grown in popularity [2]. This method relies on a human to clearly repeat the speech of interest to an automatic speech recognizer in a controlled environment.

While rapid transcription is possible with stenographers or re-speaking, these methods may not be suitable for "unstructured" recordings. These are recordings consisting of natural, spontaneous speech as well as extended periods of non-speech. BlitzScribe is designed for offline processing of unstructured audio, producing phrase-level aligned transcripts by combining both automatic and manual processing. Our interest in such unstructured recordings is driven by our work studying early language acquisition for the Human Speechome Project (HSP) [3]. Language acquisition research has typically relied on manual transcription approaches, but unfortunately transcription times of ten to fifty times the actual audio duration are not uncommon [4–6]. This may be acceptable for small corpora, but will not scale to massive corpora such as the Speechome corpus, which contains more than 100,000 hours of audio.

1.2. The Human Speechome Project

The goal of HSP is to study early language development through analysis of audio and video recordings of the first two to three years of one child's life. The home of the family of one of the authors (DR) with a newborn was outfitted with fourteen microphones and eleven omnidirectional cameras. Ceiling mounted boundary layer microphones recorded audio at 16 bit resolution with a sampling rate of 48 KHz. Due to the unique acoustic properties of boundary layer microphones most speech throughout the house, including very quiet speech, was captured with sufficient clarity to enable reliable transcription. Video was also recorded throughout the home to capture non-linguistic context. Our current analysis of the Speechome data is on the child's 9–24 month age range, with our first results reported in [7]. However, beyond our analyses of the Speechome corpus, we hope to contribute new tools and methods for replicating such efforts in the future. The remainder of this paper describes BlitzScribe, our system for rapid speech transcription.

2. Semi-automatic Speech Transcription

Functionally, manual speech transcription can be divided into four basic subtasks:

1. FIND the speech in the audio stream.
2. SEGMENT the speech into short chunks of speech.

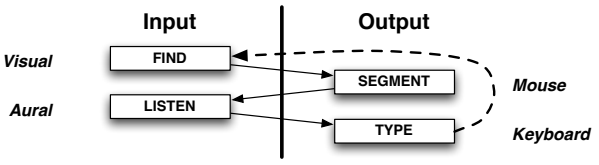

Figure 1: *Functional decomposition of manual transcription.*

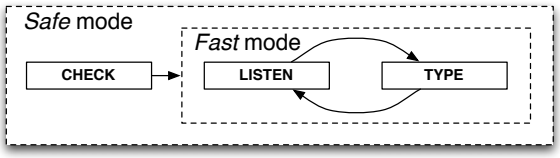

Figure 2: *User interaction model for BlitzScribe, which breaks the FSLT cycle and introduces an optional CHECK step.*

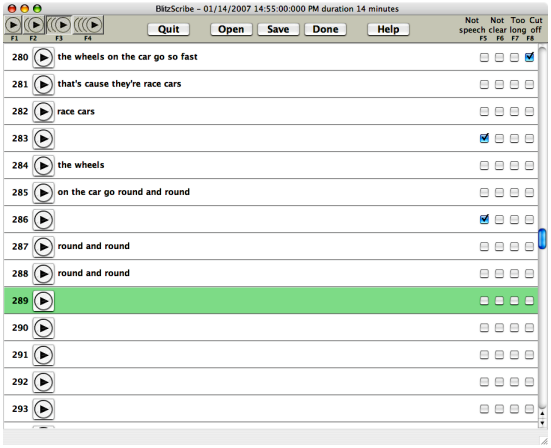

Figure 3: *The BlitzScribe user interface. Here the transcriber is listening to segment 289, highlighted in green.*

3. LISTEN to the speech segment.
4. TYPE the transcription for the speech segment.

Figure 1 depicts the FSLT sequence, along with the modality of interaction at each stage. For example, FIND is primarily a visual input task. Most transcription tools display a waveform or spectrogram to facilitate finding speech. The user visually scans the waveform or spectrogram, essentially querying the interface for the next region of speech. In contrast, SEGMENT requires output from the user, usually via the mouse. The user then listens to the segment and types a transcript. Often, a segment must be replayed to find good segment boundaries. This FSLT sequence is a reasonable sketch of the transcriber's task using either CLAN [8] or Transcriber [4], two popular transcription tools. One criticism of this approach is that it relies on an inefficient user interaction model – the user constantly switches between physically separated input devices (keyboard and mouse). It also requires the user engage in rapid *context switching*, altering between visual and aural sensory modalities, input and output subtasks, and interaction modes (textual vs. spatial). The cost of this cycle both in terms of transcription time and user fatigue is high.

In stenography, the stenographer uses only a keyboard interface and need not browse an audio stream. In other words, dispensing with the FIND and SEGMENT tasks in Figure 1. Figure 2 illustrates our design goal – a streamlined system which focuses human effort where it is necessary, and replaces the identification and segmentation of speech with an automatic system. This leads to a simple user interface, eliminating the need for the mouse and the associated costs of physically moving the hands between devices.

2.1. The BlitzScribe transcription system

There are two main components to the BlitzScribe system: an automatic speech detector and an annotation tool. These two components are connected via a central database, which stores the automatically identified speech segments as well as the transcriptions provided by the human annotator.

The system works as follows: the automatic speech detector processes unstructured audio and outputs a set of speech segments. For the Speechome corpus, the audio is multitrack (14 channels) so the speech detector must also select the appropriate channel. Speech segments, which are triples of start time, end time and channel, are stored in a relational database. Transcription is performed using the BlitzScribe interface, shown in Figure 3. Graphically, each speech segment is represented by a text box where the transcriber enters the transcript, and several checkboxes for indicating common error types. By using the arrow keys, or by typing a transcript and hitting "return," the user advances through the list. A segment can be replayed by hitting "tab." One common error introduced by the speech detector is the misidentification of non-speech audio as speech. These errors are handled in a natural way: with no speech to transcribe, the transcriber leaves the field blank, presses return to advance, and BlitzScribe marks the segment as "not-speech." Both the transcripts and the not-speech labels are stored in the database, and this information can be used later to improve the speech detector performance. The transcriber can also provide feedback on the segmentation quality by marking the segment as "too long" if it includes non-speech or "cut off" if starts or ends in the middle of an utterance.

False positives are quickly identified using the BlitzScribe interface. However, false negatives, or speech that has been missed by the automatic speech detector, require a different approach. To this end, we use TotalRecall [9] to find missed speech. TotalRecall was developed as a separate tool for data browsing and annotation. It presents all audio and video channels in a timeline view, displaying audio using spectrograms. Detected speech segments are overlaid on top of the spectrogram. TotalRecall can be used in a special mode that presents only the portions of the spectrogram where speech was *not* detected, since this is the audio that might contain false negatives. This reduces the amount of audio to consider and helps focus the user's attention. In this mode, missed speech can be spotted and saved to the database for transcription. We call transcription with this optional CHECK step "safe mode", and transcription without this step "fast mode." Figure 2 shows the relationship between these modes.

In the BlitzScribe system, the human annotator and the automatic speech detector are intimately linked. Speech found by the speech detector is presented to the human annotator for transcription. The process of transcribing provides feedback to the automatic system; each segment is effectively labeled as speech (if there is a transcript) or non-speech. This information can be used to improve the performance of the speech detector, as described in the next section.

2.2. Automatic speech detection

The automatic speech detector processes audio and outputs speech segments, which are stored in a central database. The first stage is to represent the audio as a sequence of feature vectors. Since the audio in the Speechome corpus is multitrack, the "active" audio channels are identified prior to speech detection. The resulting audio stream is then downsampled to 8 KHz (from 48 KHz in our case) and partitioned into a sequence of 30 ms frames, with a 15 ms overlap. The feature vector computed from each frame consists of MFCCs, zero crossings, power, the entropy of the spectrum and the relative power between the speech and full frequency bands. To compute the MFCCs we use the Sphinx 4 [10] libraries. The feature vectors are then presented to a "frame level" classifier, which classifies each frame as silence, speech or noise. The frame level classifier is a boosted decision tree, trained with the Weka machine learning library [11]. The frame level classifier returns a label and a confidence score for each frame.

To produce segments suitable for transcription, the sequence of classified speech frames must be smoothed and grouped together into segments. Smoothing refers to the process of relabeling frames based on neighboring frames, to help eliminate spurious classifications and produce segments of reasonable length. This is accomplished using a dynamic programming scheme which attempts to find the minimum cost labeling subject to two costs: a cost for relabeling a frame and a cost for adjacent frames having different labels. Varying the costs changes the degree of smoothing. Segmentation is accomplished simply by identifying groups of smoothed speech frames. However, speech segments which are too long are split into shorter segments at points of minimum energy, or at "low confidence" points. Confidence is lower where unsmoothed frame labels were non-speech, or the original frame classification confidence was low.

To train the speech detector, we first fetch human labeled segments from the database and apply feature extraction. This yields a training set, which is used to train the boosted decision tree frame-level classifier. The smoothing parameters (the opposing costs for state switching and relabeling) can also be learned, though in practice we have simply selected these values by hand.

3. Evaluation

In this section, we present an empirical evaluation of the system performance, showing BlitzScribe to be between 2.5 and 6 times faster than CLAN and Transcriber, two popular transcription tools. We also consider inter-annotator consistency to ensure that we are not compromising quality for speed.

3.1. Transcription speed comparison

We first measured transcription times for the HSP data using CLAN and Transcriber to form a baseline. To make the comparison fair, we assumed that the multitrack audio had been preprocessed into a single audio stream. The experimental procedure was to ask multiple transcribers to separately transcribe the same blocks of audio using the same tool, and to record the time it took to complete the task. Six audio clips were exported from the HSP corpus, five minutes each, from different times of day in two separate rooms. These blocks contained significant speech activity. Before beginning with a tool, transcribers had practiced using the tools on other audio clips.

CLAN was used in "sonic mode," in which the waveform is displayed at the bottom of a text editor. The user transcribes by highlighting and playing a block of audio, typing a transcription, and then binding the transcription to the audio segment using a key combination. In Transcriber, key combinations can be used to play or pause the audio. A segment break is created whenever the user hits return. The segment break ends the previous segment and begins a new one. Each segment corresponds to a line in the transcription window. By typing, a transcription is entered which is bound to the current segment. For this evaluation, we sidestepped the issue of speaker annotation. In CLAN, this is part of the process, in Transcriber it adds significant overhead, and in BlitzScribe speaker identity is annotated automatically, but must be checked and corrected using a separate tool.

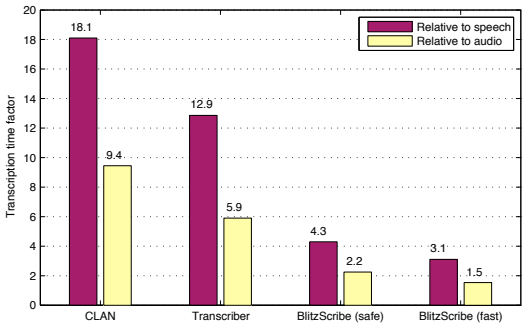

Figure 4: *Transcription time factor comparison*

To evaluate BlitzScribe, we began by considering fast mode, and then evaluated the additional time required for the CHECK step of safe mode. The experimental setup was essentially the same as above, with blocks of HSP data assigned to the same three transcribers. We recorded their times on audio of 5, 10 and 15 minutes in length. The time to perform CHECK in TotalRecall was recorded relative to the audio duration and the "non-speech time," which is the total audio duration minus the duration of just the speech. The speech duration is the cumulative duration of the speech segments. The non-speech time is of interest because it represents the amount of spectrogram the annotator must inspect.

Figure 4 summarizes the results, showing the average transcription time-factors relative to the audio duration and the speech duration. We found Transcriber to be somewhat faster than CLAN, requiring about 6 times actual time, or 13 times speech time (though performance suffers when speaker identity is annotated.) CLAN requires about 9-10 times actual time, or 18 times speech time. BlitzScribe in fast mode required about 1.5 times actual audio time, and 3 times speech time. The cost of the CHECK step was about .71 times the audio duration, and 1.3 times the non-speech time. Adding this step to fast mode transcription results in a time-factor of about 2.25 for actual audio duration and 4.3 times speech time for safe mode.

These measurements raise an interesting question: how do the two factors of audio duration and speech duration affect transcription time? The CLAN and Transcriber user interfaces entangle these two factors, since the audio duration determines how much data there is to browse, while the speech time determines the amount of annotation required. We found a very consistent linear relationship between speech time and transcription time in BlitzScribe. On the other hand, the CHECK step relationship appeared non-linear, and in [12] we explored a power-law model to account for browsing and annotation times.

In this work, we also explored the relative cost of identifying and correcting false negatives (the CHECK step) to the cost incurred by false positives. We then used the relative costs to tune the speech detector to optimize performance and minimize the overall transcription time. Overall, the transcription speed is consistent with the range reported in [6].

3.2. Transcription accuracy evaluation

BlitzScribe is designed for fast orthographic speech transcription, with phrase-level alignment between the transcripts and the audio segments. An accurate transcript is one which faithfully captures the words uttered in a given speech segment. In order to obtain accuracy measures in the absence of a ground-truth reference transcript, we look instead at inter-transcriber agreement. Our assumption is that when multiple transcribers agree, they have (likely) converged on the correct transcription.

To evaluate accuracy, we used the NIST "sclite" tool [13], which takes a set of reference and hypothesis transcripts and produces an error report. Accuracy between a reference transcript R and hypothesis transcript H is simply the fraction of correct words in H relative to the total number of words in R. Lacking a reference transcript, we calculated a symmetric accuracy assuming first one transcript and then the other to be the reference, then averaging. With this framework, we calculated the symmetric accuracy for seven transcribers using BlitzScribe on a large number of transcripts for an unfiltered audio set, and a second calculation over a smaller set of transcripts for "cleaner" audio. The average number of overlapping words per transcriber pair was about 3700 words for the larger set, and 1000 words for the smaller, filtered set. We obtained an average pairwise accuracy of about 88% and 94% for these two sets, respectively.

It was only after inspecting the transcription errors for the first set that we realized just how challenging "speech in the wild" is to transcribe. The Speechome corpus contains natural speech in a dynamic home environment, with overlapping speech, background noise and other factors that contribute to a difficult transcription task. Even after careful listening, the authors could not always agree on the best transcription. This point was also noted in [14]. Therefore, our second evaluation focused on a subset of audio which was mostly adult speech, such as when an adult was talking with the child at mealtime. Many of the errors we did observe were for contractions and short words such as "and," "is" and so on. This is comparable to the findings in [15]. Perhaps unique to our method, there were some errors where a word at the end of a segment was transcribed in the subsequent segment. While this was rare and for our purposes, not an egregious error, it was penalized nonetheless. Overall, we find that the transcription accuracy with our system and the issues we encountered are very similar to those observed in [14].

4. Conclusion

Automating the FIND and SEGMENT subtasks of traditional manual speech transcription leads to significant speed gains. In practice, BlitzScribe is between 2.5 and 6 times faster than two popular manual transcription tools. This is partly because finding and segmenting speech is inherently time consuming. Breaking the FSLT cycle reduces cognitive load, and eliminating the mouse allows the user to keep their hands in one place and focus on listening and typing. This is where human expertise is needed, since many interesting transcription tasks contain challenging speech which is difficult for human transcribers and impossible for today's automatic speech recognizers.

We have introduced BlitzScribe as a semi-automatic speech transcription system, which we have been using to transcribe the Speechome corpus for the past two years. Our current team of 14 transcribers have average a transcription speed of about 800-1000 words per hour, with some peaking at about 2500 words per hour. Collectively, they transcribe about 100,000 words per week, focusing on the child's 9-24 month age range. We expect this subset of the corpus to contain about 10 million words. Using traditional methods, transcribing a corpus of this size would be too time consuming and costly to attempt. With BlitzScribe, we have transcribed close to 30% of this data, which is already providing new perspectives on early language development.

5. References

[1] M. P. Beddoes and Z. Hu, "A chord stenograph keyboard: A possible solution to the learning problem in stenography," *IEEE Transaction on Systems, Man and Cybernetics*, vol. 24, no. 7, pp. 953–960, 1994.

[2] A. Lambourne, "Subtitle respeaking: A new skill for a new age," in *First International Seminar on New Technologies in Real Time Intralingual Subtitling*. inTRAlinea, 2006.

[3] D. Roy, R. Patel, P. DeCamp, R. Kubat, M. Fleischman, B. Roy, N. Mavridis, S. Tellex, A. Salata, J. Guinness, M. Levit, and P. Gorniak, "The human speechome project," in *Proceedings of the 28th Annual Cognitive Science Conference*, 2006, pp. 2059–2064.

[4] C. Barras, E. Geoffrois, Z. Wu, and M. Liberman, "Transcriber: Development and use of a tool for assisting speech corpora production," *Speech Communication*, vol. 33, no. 1-2, pp. 5–22, January 2001.

[5] D. Reidsma, D. Hofs, and N. Jovanović, "Designing focused and efficient annotation tools," in *Measuring Behaviour, 5th International Conference on Methods and Techniques in Behavioral Research*, Wageningen, The Netherlands, 2005.

[6] M. Tomasello and D. Stahl, "Sampling children's spontaneous speech: How much is enough?" *Journal of Child Language*, vol. 31, no. 1, pp. 101–121, February 2004.

[7] B. C. Roy, M. C. Frank, and D. Roy, "Exploring word learning in a high-density longitudinal corpus," in *Proceedings of the 31st Annual Cognitive Science Conference*, In press.

[8] B. MacWhinney, *The CHILDES Project: Tools for Analyzing Talk*, 3rd ed. Lawrence Erlbaum Associates, 2000.

[9] R. Kubat, P. DeCamp, B. Roy, and D. Roy, "TotalRecall: Visualization and semi-automatic annotation of very large audio-visual corpora," in *ICMI*, 2007.

[10] W. Walker, P. Lamere, P. Kwok, B. Raj, R. Singh, E. Gouvea, P. Wolf, and J. Woelfel, "Sphinx-4: A flexible open source framework for speech recognition," Sun Microsystems, Tech. Rep. 139, November 2004.

[11] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed., ser. Series in Data Management Systems. Morgan Kaufmann, June 2005.

[12] B. C. Roy, "Human-machine collaboration for rapid speech transcription," Master's thesis, Massachusetts Institute of Technology, Cambridge, MA, September 2007.

[13] J. Fiscus. (2007) Speech recognition scoring toolkit ver. 2.3 (sctk). [Online]. Available: http://www.nist.gov/speech/tools/

[14] J. Garofolo, E. Voorhees, C. Auzanne, V. Stanford, and B. Lund, "Design and preparation of the 1996 Hub-4 broadcast news benchmark test corpora," in *Proceedings of the DARPA Speech Recognition Workshop*, 1996.

[15] W. D. Raymond, M. Pitt, K. Johnson, E. Hume, M. Makashay, R. Dautricourt, and C. Hilts, "An analysis of transcription consistency in spontaneous speech from the buckeye corpus," in *ICSLP*, 2002, pp. 1125–1128.