Audio Puzzler: Piecing Together Time-Stamped Speech Transcripts with a Puzzle Game

Nicholas Diakopoulos, Kurt Luther, Irfan Essa
Georgia Institute of Technology | School of Interactive Computing | GVU Center
85 5th St. NW Atlanta, GA 30332
nad@cc.gatech.edu, luther@cc.gatech.edu, irfan@cc.gatech.edu

ABSTRACT
We have developed an audio-based casual puzzle game which produces a time-stamped transcription of spoken audio as a by-product of play. Our evaluation of the game indicates that it is both fun and challenging. The transcripts generated using the game are more accurate than those produced using a standard automatic transcription system and the time-stamps of words are within several hundred milliseconds of ground truth.

Categories and Subject Descriptors
H.5.2 [**User Interfaces**]: Graphical user interfaces

General Terms: Design, Human Factors

1. INTRODUCTION
Audio Puzzler (http://www.audiopuzzler.com) is a casual puzzle game which uses real spoken audio clips taken from short videos as puzzle pieces. Players must first transcribe and then piece together snippets of these audio clips in order to complete the puzzle. Audio Puzzler is designed to be fun and enjoyable to play while also producing meaningful output in the form of time-stamped transcriptions of the audio used for the puzzle. The method of reformulating a task as a game in order to get work done while people play is known as "Human Computation" and has been successfully applied in areas such as image and music tagging [11, 12].

Accurate transcriptions are valuable for improving information access, searchability, and browsability of audio or video clips [7, 10], but current Automatic Speech Recognition (ASR) systems typically produce Word Error Rates (WER) of 20% to 45% in real world conditions [7]. Only in the best of acoustic conditions and with training for a particular speaker can the WER get much lower. Furthermore, computing accurate time stamps for individual words from imperfect ASR transcripts is an area of current research in multimedia. Some recent results align words to within 10 seconds of their actual occurrence in the stream [2]. Having more precisely aligned transcripts enables new types of multimedia interfaces which tightly couple the transcript to the timeline of a video.

Our system approaches the time-stamped speech transcription problem using a lusory (game-based) human computation method. We evaluated our system (1) by conducting a user study to assess the enjoyability of the game aspects and (2) by comparing the transcriptions and time-stamps generated using the game to manually labeled ground truth transcripts. In the following sections we present the design, algorithms, and evaluation of the system and conclude with a discussion of the results.

MM'08, October 26–31, 2008, Vancouver, British Columbia, Canada.

2. AUDIO PUZZLER

2.1 Game Design
Audio Puzzler is a new type of Flash-based online puzzle game that uses speech audio as the basis for the puzzle pieces. The goal of the game is to assemble the puzzle pieces as quickly and accurately as possible. At the outset, each puzzle starts as a set of audio bubbles (Figure 1a.), which are grouped by color to simplify the puzzle (Figure 1). Double-clicking an audio bubble pops the bubble and plays the audio in that bubble. The interface then allows the player to type the words that were spoken in the audio (Figure 1b). Audio can be repeated by clicking an icon or pressing a key combination. Once typed, the textual puzzle pieces can be dragged on top of one another to connect them. If the text matches and they belong next to each other, then the matching words are highlighted in green (Figure 1c). Dropping the piece merges them. If, for some reason, the words should match but do not (e.g., because of a typographical error), then they are highlighted in red (Figure 1d). As pieces are assembled, progress is shown at the top by filling in the colored meters. Time and score as well as a pause button are also shown in the top area. Upon completing the puzzle, the player can optionally listen to and watch the video from which the audio was taken.

Each audio puzzle consists of 3 levels of increasing difficulty, each with more puzzle pieces to assemble. The amount of audio in each level roughly corresponds to 15s, 20s, and 25s. For each level, the audio is broken into a number of overlapping audio chunks represented as bubbles. The overlap is essential so that the player can see where the puzzle pieces match up once they are typed. Bubbles containing only silence (according to a simple short-term energy feature) are filtered out of a level.

The scoring system in Audio Puzzler is designed to promote quick completion of the puzzles and to make the game more challenging by providing a sense of time pressure. The game clock starts off with 3 minutes and counts down. Each time the player successfully merges two pieces they get more time on the clock. The amount of time added is about 17 seconds and was determined through early play testing so as not to make the game too easy or difficult. Also, when the player merges two pieces they get a number of points proportional to how early in the level the merge took place. Merges earlier in the level receive more points than merges later on. This encourages connecting the pieces and completing the puzzle as quickly as possible. Additionally, bonus points are awarded at the end of the level

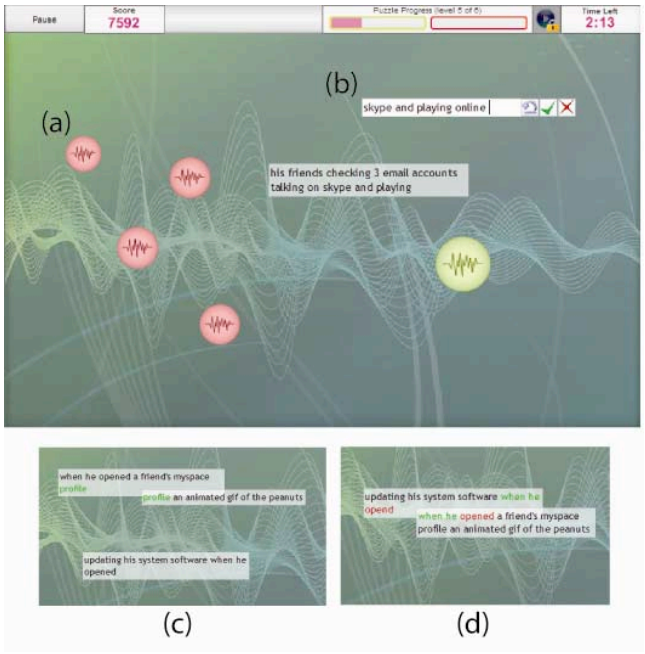

Figure 1. Overview of the Audio Puzzler interface.

based on how quickly the level was completed. If the time clock falls below zero, the game continues (as completion produces more transcription), but the player is penalized 5 points every 5 seconds until the clock rises above zero again. This, too, lends a sense of pressure and urgency to the game. At the end of the game, the player receives another bonus (or penalty) based on the time left on the clock.

Care was taken in the design of Audio Puzzler to adhere to good heuristics of game design such as providing adequate feedback, control, and cognitive engagement [1, 4, 8]. Animation and color are used throughout the game to make it more "juicy" to play [9]. Bubbles bob and pulse, hearts shoot from merged pieces, and large text animations mark the completion of color groups and levels. We also tried to balance the fun of the game against the underlying goal of generating good transcripts by setting parameters such as chunk length, overlap, and game length.

2.2 Transcription Algorithm

Before text from audio chunks is merged to form the transcript, it is normalized using a set of filters [3] so that comparisons to the ground truth are more fair. Contracted words such as "we're," "They're" etc. are expanded to "we are," and "they are", respectively. Also, numbers written with numerals are expanded so that, for example, "2008" becomes "two thousand and eight."

At the completion of a game, we have a set of n audio chunks C_1 … C_n with their transcriptions ordered by start time. Chunk duration is chosen randomly at the start of the game to be 2.25s – 2.50s and each chunk has a .75s overlap with the chunk preceding it. These parameters were chosen to facilitate the user only having to listen to each chunk once during game play as well as to ensure sufficient word overlap between pieces. By looking at the average spoken word rate for English (about 160 words per minute or about 2.67 words per second [13]) in conjunction with the short term memory limits of humans (about 7 chunks [6]), we determined that an audio chunk length of about 2.5 seconds would lead to about 7 words per chunk. Thus, the player should be able to keep these 7 words in short term memory during transcription and avoid having to replay the audio.

To merge the audio chunks we take each pair of adjacent chunks C_i and C_{i+1}, for i = 1 to n-1, and convolve the words of C_i, with C_{i+1}. If two words match then the score at that point in the convolution is incremented. The highest score is taken as the optimal overlap point. The words before this point are taken from C_i and the words after this point are taken from C_{i+1} as shown in Figure 2. The best overlap point is computed with,

$$\text{Max index}\left[\sum_{j=0)}^{|C_i|} \sum_{k=0}^{|C_{i+1}|} C_i^{j+k} \cdot C_{i+1}^{k} \right]$$

Figure 2. Merging two audio chunks with the optimal overlap point shown in solid red

To compute time stamps for each word, we make the assumption that over the course of a chunk, words and pauses occur at a steady rate. Better approximations to the duration of each word and to inter-word pauses could be made using TTS (Text to Speech) techniques. Since we know the beginning and ending time of each audio chunk, each word is assigned a beginning and ending time by equally dividing the chunk duration by the number of words in the chunk taking into account inter-word pause duration, which we set at 20ms. For words that occur in multiple chunks (e.g., the last two words in C_i shown in Figure 2) the time-stamps are estimated and averaged from both chunks. If the end time of any word follows the start time of any word, we assign the average of the two as the end of one and the start of the other. This ensures that word time stamps do not overlap.

3. EVALUATION

Our high-level measure of success was a game that is fun to play and also produces valuable transcripts. To evaluate Audio Puzzler, we undertook both (1) a user study to assess aspects of the enjoyment and usability of the game and (2) an analysis of the accuracy of transcripts and time-stamps output from the game.

3.1 Audio Puzzler Interface

Based on game design heuristics, the user study was developed to assess the game's enjoyability and fun, challenge, replayability, pacing, and usability in terms of feedback, game status, and controls [1, 5, 8]. We first ran a pilot study with 5 participants to understand most of the usability issues and iterate on the interface until it was stable and usable. We then ran the main study.

Playtesting of Audio Puzzler involved 3 methods: observation, questionnaires, and logging. Observation of players was used to identify reactions such as smiles and laughter, cognitive difficulties or confusions with the controls, and content effects such as indications of engagement with or comprehension of the audio content. A questionnaire was used to collect self-reported information about participant background, fun, replayability, usability, pacing, and challenge. User scores, elapsed time, and the transcriptions were also logged for analysis.

Ten people participated in the main user study (9 male, 1 female). Eight participants were native speakers of English and 2 were not.

We first explained the instructions for the game and briefly demonstrated the controls to participants. Next, we asked each participant to play the game twice, once each with a different audio puzzle. In total there were 3 puzzles used in the playtesting; each participant played with 2 out of the 3. The order of the presentation of the puzzles was counterbalanced across participants. After the second game, participants filled out the questionnaire and were informally interviewed.

The content of the three audio puzzles was selected for understandability from a set of five video clips used in the pilot. We found in the pilot that if the speaker in the video clip used awkward grammar or spoke too slowly this affected the playability of the game. The final content consisted of a news clip about medical error, a snippet of a speech made by Al Gore taken from *An Inconvenient Truth*, and a portion of a fake comedic newscast from *The Onion*. Each clip was 60s to 70s long.

Summary statistics for fun, challenge, and pacing from the questionnaire and game time from the log analysis are shown in Table 1. The numbers show that participants found the game to be fun (4.65), but too challenging (3.3) and too long (5.75). Some participants laughed and smiled while playing the game. Others were vocal, making comments like, *"yay," "that was fun,"* or *"yes, ohh boy"* after completing a level or game, or just finding matching pieces. Several participants said that the idea of a puzzle involving audio was interesting and that trying to figure out how to put the puzzle pieces together was engaging and fun. One wrote, *"I liked the challenge of anticipating where different clips were in the story line."* A few participants mentioned competition and getting the high score as another fun aspect.

Content was a significant component of what was fun or not about the game, with participants indicating that the two informational clips, while informative, were less fun to play with than the *Onion* clip. Although this is supported by quantitative responses on the questionnaire, the difference was small; the average fun for the informative clips was 4.62 and the average fun for the onion clips was 4.71. Others liked the challenge of multitasking (i.e., listening, typing, and scanning for matches) or the challenge of the time pressure. Several participants mentioned that they liked the content they played with whether that be informative or entertaining; one participant appreciated being able to learn from the content of the game. Among the things that participants disliked about the game were that a social element (either competitive or cooperative) was missing and that there was too much typing involved in the game.

Some players commented that spelling was an issue. It was sometimes difficult to hear words because they were truncated at the beginning or ending of a chunk, and the game was more difficult if you were not a good typist. One participant thought it was too cognitively demanding to play, writing, *"It took a lot to listen to the audio and then type it and also think in the back of my mind about what I've heard so that I can make connections as soon as I was done typing the audio."*

	Average	Std Dev
Fun (1 not fun, 7 fun)	4.65	1.18
Challenge (1 too hard, 7 too easy)	3.3	1.06
Level Duration (1 too short, 7 too long)	5	.48
Game Duration (1 too short, 7 too long)	5.43	1.25
Game time (seconds)	875 s	23.90 s

Table 1. Summary statistics for game play ratings.

During our observations of game play, we took special note of the effect of time pressure. We found that some participants appeared unaffected by the time pressure and enjoyed the game for the sake of solving the puzzle. Others were fixated on time and with being the fastest and best. This mentality led to play strategies which involved the use of abbreviations, numerical representations of numbers, and the elision of some words for each piece to reduce typing. These participants saw that there were usually multiple words overlapping between adjacent audio pieces and learned that they only needed one overlapping word to make a match. One participant went so far as to try transcribing only the first and last words of each piece, but soon found that this was ineffective for rapidly finding matches and reverted to typing out most words.

We observed how important context is for understanding individual words and for transcription in general. In several cases, participants had trouble understanding particular words or phrasing of isolated pieces. For example, a participant first transcribed a word as "gift" but upon hearing the context of the word when another piece was played corrected the transcription to "animated .gif." Participants would often go back and correct transcriptions of pieces as they heard more audio pieces and better understood the context of previous pieces.

3.2 Transcription Accuracy

The second goal of our evaluation was to analyze the accuracy of the transcripts and time stamps produced. We created a ground truth transcript with time stamps of each word for the 3 audio puzzles. One of the authors manually marked the beginning and ending time of each word in the clip, a tedious process requiring about 3 hours to transcribe 1 minute of audio.

We compared each transcript produced by a participant using the game to (1) the ground truth and (2) automatically generated transcripts produced using the Sphinx 3 speech recognizer[1]. We computed the Word Error Rate (WER) [3] in comparison to both the ground truth and ASR output. We also computed the mean error for the time stamps of words that were aligned between the hypothesis transcript and the reference transcript using the following equation,

$$\sum_{T_i=\hat{T}_j} .5 \cdot \left(\left| S_i - \hat{S}_j \right| + \left| E_i - \hat{E}_j \right| \right)$$

Where S_i is the start time of the i^{th} word in the hypothesis transcript and $\hat{S}_j$ is the start time of the j^{th} word in the reference transcript; E_i and $\hat{E}_j$ refer to the time stamps for the end of the word. T_i refers to the i^{th} word in the hypothesis transcript and $\hat{T}_j$ refers to the j^{th} word in the reference transcript.

Summary statistics for WER and time stamp accuracies are shown in Table 2. The average WER rate for the three audio clips used was 10.08%, much lower than the 53.76% WER resulting from the ASR transcripts. The word time stamps generated using the game output were on average within .373s of the actual time-stamp from the ground truth. Based on the amount of time

[1] http://cmusphinx.sourceforge.net/html/cmusphinx.php

participants took to produce a transcript, we also computed the average play time need to produce each word as 4.95s.

	Average	Std Dev
WER (game output vs. ground truth)	10.08%	6.60
WER (ASR vs. ground truth)	53.76%	10.85
Time stamp (game output vs. ground truth)	.373 s	.108 s

Table 2. Summary statistics for transcription accuracy.

4. DISCUSSION AND FUTURE WORK

Literature suggests that even a 25% WER for a transcript is acceptable to start enhancing information searching and seeking behavior in multimedia systems [7, 10]. Thus, Audio Puzzler's WER of 10.08% is a success. Informal analysis of the transcripts revealed that many of the errors produced using Audio Puzzler are the result of misspellings or typographical errors. This suggests that WER results can be further improved by running transcripts through a spell checker. In the future we hope to fuse the data from multiple players of the game to aggregate even better transcripts. When a group of people play the game with the same puzzle, redundancies may lead to even lower WER results.

The other factor that affects transcription accuracy is the background, language ability, and knowledge of the player. There was a negative correlation (-.55) between the number of years our participants had lived in an English speaking country and the WER of their transcriptions. While some of this effect is clearly due to language ability, another part of it may be cultural knowledge. In the *Onion* clip there were many references to popular culture (e.g., bands, singers, products, etc.) that may have been more difficult for players to recognize and transcribe if they did not have cultural familiarity with those named entities.

We found Audio Puzzler to be moderately fun for participants based on the puzzle aspect of the game, competition, and scoring. There was a strong self-reported content effect and when asked if they would play the game again, several participants mentioned that content (and choice of content) would be a key determinant. While comedic content appealed more than informative content in general, participants suggested all sorts of different content that would appeal to them individually, including song lyrics, standup comedy, or soft news. A couple of participants mentioned that they would be more motivated to play the game if they played with personally relevant content, such as political speeches, knowing that the output of the game would be a transcription of the speech from which others might benefit.

This content effect puts limitations on the type of content that can be effectively transcribed with the system. Engaging content is more fun to transcribe. More entertaining content such as comedy is likely to be actively played, whereas informative content such as news or documentary may appeal to a smaller audience. Boring, irrelevant, or slow audio such as meeting recordings are unlikely to be compelling sources of content for the game. In other words, even though an arbitrary piece of audio *could* theoretically be used to create a puzzle for the game, in practice, some editorial decision needs to be made by a human to, at the very least, select appealing content. Furthermore, in order to keep the duration of the game reasonable, the application is best suited for short pieces of audio from sources such as YouTube videos or short news pieces.

Based on the user study, the game still needs to be tweaked so that it is shorter and easier. The duration of the game can be manipulated by either forming more levels of shorter length, or by simply using shorter audio segments as input for the puzzles. One comment made by several of the participants was that hearing partial words at the beginning or ending of audio pieces was confusing and added to the difficulty of the game. In the future, we would like to compute audio pieces so that their beginning and ending times fall between words. This should make the listening component of the game easier for players.

5. CONCLUSIONS

We have demonstrated that a game-based approach toward time-stamped transcript generation is effective in producing accurate transcripts and time-stamps and in providing for a challenging, engaging, and fun experience for players. Players found the puzzle aspect of the game engaging, but parameters for challenge and game duration need to be further tweaked. While content choice generally affects what creates a meaningful and fun playing experience, there is much variability in individual preference for content. With enough content, Audio Puzzler could appeal to a wide variety of people across the internet.

6. ACKNOWLEDGMENTS

Many thanks to our playtesters especially Kathryn Bergman and Dimitri Diakopoulos as well as to Nikolaos Vasiloglou.

7. REFERENCES

[1] Desurvire, H., Caplan, M. and Toth, J., Using heuristics to evaluate the playability of games. in *Proceedings of CHI*, (2004), 1509-1512.

[2] Haubold, A. and Kender, J.R., Alignment of Speech to Highly Imperfect Text Transcriptions. in *International Conference on Multimedia and Expo (ICME)*, (2007), 224-227.

[3] Huang, X., Acero, A. and Hon, H.W. *Spoken Language Processing*. Prentice Hall, 2001.

[4] Koster, R. *A Theory of fun for game design*. Paraglyph, 2005.

[5] Malone, T., Heuristics for designing enjoyable user interfaces: Lessons from computer games. in *Proceedings of CHI*, (1982), ACM, 63-68.

[6] Miller, G. The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *The Psychological Review*, *63*. 81-97.

[7] Munteanu, C., Baecker, R., Penn, G., Toms, E. and James, D., The Effect of Speech Recognition Accuracy Rates on the Usefulness and Usability of Webcast Archives. in *Proceedings of CHI*, (2006), 493-502.

[8] Pinelle, D., Wong, N. and Stach, T., Heuristic Evaluation for Games: Usability Principles for Video Game Design. in *Proceedings of CHI*, (2008), 1453-1462.

[9] Shneiderman, B. Designing for fun: how can we design user interfaces to be more fun? *interactions*, *11* (5). 48-50.

[10] Stark, L., Whittaker, S. and Hirschberg, J., ASR Satisficing: The effects of ASR accuracy on speech retrieval. in *Proceedings of International Conference on Spoken Language Processing*, (2000).

[11] Turnbull, D., Liu, R., Barrington, L. and Lanckriet, G., A Game-Based Approach for Collecting Semantic Annotations of Music. in *International Symposium on Music Information Retrieval*, (2007).

[12] von Ahn, L. and Dabbish, L., Labeling images with a computer game. in *Proceedings of CHI*, (2004), 319-326.

[13] Yuan, J., Liberman, M. and Cieri, C., Towards an Integrated Understanding of Speaking Rate in Conversation. in *Proceedings of the Conference on Spoken Language Processing*, (2006).