

# **Remembering 9/11 – Accessing Oral Histories for Educational and Research Purposes**

**Ryan Goepfert**  
Tony Scarlatos

## **Introduction**

The Stony Brook WTC Wellness Program provides support to 9/11 responders, treating illnesses of both body and mind. The program mission aims to provide “an enduring support community, advocacy for our 9/11 responders, and cutting edge research efforts” [1]. Part of this research has included conducting interviews with responders with the intent to analyze them, provide access to all interviews and transcripts to the public, and create an environment where educators could conduct lessons with them.

## **Problem**

The program was in need of a way to display such a large collection of interviews and related transcripts in a form that allowed easy access and navigation for the public, researchers, and educators. The program wished to display both the video interview and transcript at the same time, allowing a wider range of viewing options. With a collection of interviews totaling over a hundred, search functionality and other organization features were needed for ease of use. A way to annotate the videos was also needed for researchers to easily collaborate on finding the main themes common among responders.

## **Research**

I first looked into how to display the transcripts along with the video. Closed captioning was the first thought that came to mind. I wrote a script to convert the transcripts to several closed captioning formats for easy testing. Upon receiving one of the interview transcripts, however, it was clear that this option would not suffice, as the sections of text between each timestamp were much too large; the text would cover most of the video and it did not always fit inside of the video frame. Closed captioning would not work unless all the transcripts were modified; the program already paid for the transcription service and liked the context given by showing a larger section of transcript at any given time.

I then looked into services that could render the transcript as text on the page while the video played above it. The relevant section of text would highlight when the video got to that point. These services were costly and were packaged as part of a transcription service, something that the program already had complete. The transcript would also be below the video, which would not allow users to view both at the same time.

General video hosting services were then considered, with special interest on those that could display two videos side-by-side. This could allow me to convert the transcript into a video, displaying the appropriate sections of text as the interview hit the timestamps. Services were again expensive, generally lacked side-by-side video support, or non-open source, making it difficult to add needed features and customize as needed.

Services that provided annotations were researched as well. An anthropologist working on the project had created a coding system for annotating the interviews. An annotation system that allowed for codes when adding annotations was needed to more easily create and review annotations.

## **Solution**

A system that seemed to have everything required was finally found: an open-source video hosting software called Opencast [2]. Originally designed for educational institutions so lecturers could upload their lectures along with their PowerPoint slides, I was able to adopt it to show the interviews and PowerPoint slides of the transcript. There is also an add-on that allows one to annotate the uploaded videos [3]. Opencast supports search, assigning videos to a series, user permissions, and detecting text in PowerPoint uploads for use in searching, segmenting the video based on slide transitions, and to display under the video. Being open-source meant that any needed features in the future could potentially be added; it also meant that there was no cost.

While the infrastructure was available, setting it up was not easy. Opencast did not officially support MacOS X servers, although there were installation instructions for it. Working with what we had, we elected to install the software on it. The free cost of open-source software came with the drawback of no official support. Problems during installation were hard to overcome due to a lack of real support; community forums were the only means of troubleshooting one could obtain. These problems were eventually overcome for both Opencast and the annotation tool add-on, leading to a fully functioning video hosting site.

Converting the transcripts to PowerPoint videos was the next challenge to overcome. The process is simple enough: put each timestamped section of text onto its own slide, set the transitions for each slide based on the difference in the timestamps, and save the slides in mp4 format. However, with over a hundred transcripts ranging from 30 minutes to 2 hours this would be a long and tedious job. I created two python scripts to automate this process. The first parsed the timestamps from the transcripts and found the time difference between each section, then it used Python-pptx [4] to create a PowerPoint with each section of text on a new slide. The library

I used lacked a way to add transition times to the slides, so I saved the time differences in a new file. Human interaction was then needed to save the PowerPoint as an xml file. This allowed my second script to insert the transition times to the appropriate slides. Human interaction was then again needed to save the PowerPoint as an mp4 video.

This process worked well, but the transcripts I received after the initial ones differed in format. This made the script break on certain transcripts, requiring user intervention to check the transcript and resulting PowerPoint slides for error.

The Stony Brook WTC Wellness Program had an agreement with the Library of Congress to store the archive of interviews there. The program had hopes of the Library displaying the videos stored there and this system of showcasing and annotating the videos was demonstrated to the Library. The Library had current work on their own system, but wanted to stay in touch. This system was also presented to staff and curators from the 9/11 Memorial Museum, with the program hoping to get the interviews featured in the museum in some fashion. The museum was years out from implementing this sort of thing and communication was left open for discussion in the future.

## **Future Work**

For further implementations of Opencast, several features are in high demand after discussions with the Wellness Program, Library of Congress, and the 9/11 Memorial Museum, such as: adding tags to videos to more easily categorize and search them, having search link to specific points in interviews, creating a moderation system for educators to employ the annotation system within their classroom, and creation of user profiles for each interviewee that is linked to their

video, transcript, and annotation tool. Each of these seems doable from the open-source code that Opencast currently has.

Treating Opencast as the prototype and proof-of-concept, however, seems like the better way to move forward. Using this program as the basis for the creation of a dedicated platform seems like a stronger play for future development. Opencast has a lot of unnecessary features that may detract from the needed ones. The system of converting transcripts to videos could then be improved; the videos take up enough space as is, and each transcript in video format takes up another large chunk of memory. A better system would be to display the interview video on one side of the screen with the first section of transcript appearing next to the video in text format. As the video moves past each timestamp, the text would update to the next section of text. Similar to the current system, except this would only require the storage of text, reducing memory space significantly. The entire transcript could be displayed below the video or on demand by the user. The clicking of a section of this text could then skip the video to that part. The annotation tool could be modified to have a nicer UI and implement a moderation system.

An idea for the museum included placing QR codes at certain exhibits that relate to something that was said in an interview. The museum patron could scan this code and be taken to the video at the exact time the topic is being discussed.

Sentiment analysis could be run on each interview as a first pass to come up with general themes that could then be expanded upon by an anthropologist. This would save time and allow lesser trained individuals to more easily annotate the interviews.

The conversion of all transcript into a single, uniform format would greatly ease the transition of transcripts to whatever format is required in the future. Whether continuing with the Opencast version of creating PowerPoint slides movies or uploading the text directly to display

aside the interview, the current mishmash of differing formats makes either of these methods more difficult than is needed. Parsing for timestamps and relevant text sections (non-header information) would benefit greatly from this, with minimal human intervention required.

Grant money has already been considered, with interest in the National Endowment of the Humanities (NEH) and the Gardiner Foundation. Grant money could be put towards running dedicated servers to host all the interviews as well as paying programmers to continue work on the system.

## Citations

- (1) "Bringing Patient-focused Care and Research to Our Heroes." *Stony Brook WTC Wellness Program*. N.p., n.d. Web. 30 Apr. 2017.
- (2) *Opencast*. N.p., n.d. Web. 30 Apr. 2017.
- (3) "Opencast Annotation Tool." *Opencast-community / annotation-tool* — *Bitbucket*. N.p., n.d. Web. 30 Apr. 2017.
- (4) "Python-pptx." *Python-pptx*. N.p., n.d. Web. 30 Apr. 2017.