**Process Documentation — Task\_06\_Deep\_Fake**

Project: Street Interviews from 2125 — Scientific Breakthroughs for the Next Century

Date: August 31, 2025

# Process Documentation — Task\_06\_Deep\_Fake  
  
\*\*Date:\*\* August 31, 2025   
\*\*Deliverable:\*\* Audio-first “street interview” set in 2125 (optional video).   
\*\*Repo target file:\*\* Place this document at the repo root as `PROCESS\_DOCUMENTATION\_Task\_06\_Deep\_Fake.md` (and/or `.docx`).

## 1) Objective & Concept  
Create an AI-generated interview where a 2125 reporter asks experts how \*\*2020s scientific breakthroughs\*\* cascaded into century-scale change. The narrative anchors to real milestones (fusion ignition at NIF, JET sustained record, AlphaFold 3 interactions, first FDA-approved CRISPR therapy, perovskite–Si tandem solar). Speculation is clearly labeled; sources are cited in the README’s “Sources” section.  
  
\*\*Why audio-first?\*\* It’s fast, free, reproducible on a laptop, and avoids paywalls/watermarks common in video-only tools.

## 2) Environment  
- \*\*OS/Hardware:\*\* macOS (Apple Silicon) — but instructions are cross-platform where possible.  
- \*\*Tools used (free):\*\*  
 - \*\*FFmpeg 8.x\*\* — for concatenation and encoding.  
 - \*\*Piper TTS (pip install piper-tts)\*\* — local neural text-to-speech.  
 - \*\*Hugging Face Piper voices\*\* — e.g., `en\_US-amy-medium` (reporter) and `en\_US-joe-medium` (experts).  
 - \*(Optional)\* \*\*Wav2Lip\*\* or \*\*SadTalker\*\* — for talking-head video from audio.  
- \*\*Project folders (repo root):\*\* `audio/`, `voices/`, `script/`, `prompts/`, `process\_log/`, `tools-notes/`.

## 3) Data & Script  
- \*\*Theme:\*\* “Scientific breakthroughs for the next century.”  
- \*\*Script files:\*\* Split into 10 turns for timing and mixing:   
 `audio/01\_reporter.txt … 10\_all.txt` (see repo for exact lines).  
- \*\*Personas:\*\* Reporter (curious), Fusion Engineer (assured), Bio-Design Researcher (thoughtful), Global Health Lead (empathetic), Climate-Tech Builder (optimistic).

## 4) Voice Models  
Download two Piper models \*\*and\*\* their matching JSON configs to `voices/`:  
- `voices/en\_US-amy-medium.onnx` + `voices/en\_US-amy-medium.onnx.json` (Reporter)   
- `voices/en\_US-joe-medium.onnx` + `voices/en\_US-joe-medium.onnx.json` (Experts)  
  
> Voice catalog: Rhasspy Piper voices on Hugging Face (browse en\_US / amy, joe).

## 5) Commands Used (reproducible)  
\*\*Install Piper (pip):\*\*  
```bash  
python3 -m pip install --upgrade pip  
python3 -m pip install piper-tts  
piper --version # should print version and usage  
```  
  
\*\*Synthesize WAVs (Reporter = Amy | Experts = Joe):\*\*  
> Tip: adjust pacing with `--length-scale`; adjust inter-sentence pause with `--sentence-silence`.  
  
```bash  
# Reporter lines  
piper -m voices/en\_US-amy-medium.onnx -c voices/en\_US-amy-medium.onnx.json --length-scale 0.95 --sentence-silence 0.15 -i audio/01\_reporter.txt -f audio/01\_reporter.wav  
piper -m voices/en\_US-amy-medium.onnx -c voices/en\_US-amy-medium.onnx.json --length-scale 0.95 --sentence-silence 0.15 -i audio/03\_reporter.txt -f audio/03\_reporter.wav  
piper -m voices/en\_US-amy-medium.onnx -c voices/en\_US-amy-medium.onnx.json --length-scale 0.95 --sentence-silence 0.15 -i audio/05\_reporter.txt -f audio/05\_reporter.wav  
piper -m voices/en\_US-amy-medium.onnx -c voices/en\_US-amy-medium.onnx.json --length-scale 0.95 --sentence-silence 0.15 -i audio/07\_reporter.txt -f audio/07\_reporter.wav  
piper -m voices/en\_US-amy-medium.onnx -c voices/en\_US-amy-medium.onnx.json --length-scale 0.95 --sentence-silence 0.15 -i audio/09\_reporter.txt -f audio/09\_reporter.wav  
  
# Expert lines  
piper -m voices/en\_US-joe-medium.onnx -c voices/en\_US-joe-medium.onnx.json --sentence-silence 0.12 -i audio/02\_fusion.txt -f audio/02\_fusion.wav  
piper -m voices/en\_US-joe-medium.onnx -c voices/en\_US-joe-medium.onnx.json --sentence-silence 0.12 -i audio/04\_bio.txt -f audio/04\_bio.wav  
piper -m voices/en\_US-joe-medium.onnx -c voices/en\_US-joe-medium.onnx.json --sentence-silence 0.12 -i audio/06\_health.txt -f audio/06\_health.wav  
piper -m voices/en\_US-joe-medium.onnx -c voices/en\_US-joe-medium.onnx.json --sentence-silence 0.12 -i audio/08\_climate.txt -f audio/08\_climate.wav  
piper -m voices/en\_US-joe-medium.onnx -c voices/en\_US-joe-medium.onnx.json --sentence-silence 0.12 -i audio/10\_all.txt -f audio/10\_all.wav  
```  
  
\*\*Concatenate to MP3 (robust filter graph):\*\*  
```bash  
ffmpeg -i audio/01\_reporter.wav -i audio/02\_fusion.wav -i audio/03\_reporter.wav -i audio/04\_bio.wav -i audio/05\_reporter.wav -i audio/06\_health.wav -i audio/07\_reporter.wav -i audio/08\_climate.wav -i audio/09\_reporter.wav -i audio/10\_all.wav -filter\_complex "[0:a][1:a][2:a][3:a][4:a][5:a][6:a][7:a][8:a][9:a]concat=n=10:v=0:a=1,aresample=async=1:first\_pts=0[a]" -map "[a]" -c:a libmp3lame -q:a 2 audio/final\_mix.mp3  
```  
  
> Why this method? The \*\*concat filter\*\* decodes inputs and avoids list-file header quirks. `aresample=async=1:first\_pts=0` keeps the audio timeline aligned.

## 6) Issues Encountered & Fixes  
- \*\*zsh “number expected” / “parse error near ')'”\*\*   
 Cause: pasted partial commands, backslashes, or stray comments.   
 Fix: use \*\*single-line\*\* Piper commands or a small `run\_piper.sh` script.  
- \*\*FFmpeg warning: “Invalid PCM packet… size 1 expected 2”\*\*   
 Cause: one WAV had a malformed/truncated header.   
 Fix A: re-encode each WAV to clean PCM (`-ac 1 -ar 22050 -c:a pcm\_s16le`) then concat.   
 Fix B (used here): \*\*concat filter\*\* one-liner (see above), which decodes and stitches safely.  
- \*\*Audio normalizing attempt produced silence\*\*   
 Cause: `afade` used with `st=0` (fade-out starting at 0 seconds).   
 Fix: skip fades for final delivery or start the fade near the end of the file. Keep `final\_mix.mp3` as the deliverable.

## 7) Ethics & Provenance  
- \*\*Disclosure:\*\* AI-generated educational demo; fictional personas; no real-person voice cloning.   
- \*\*Responsible practices:\*\* Follow Partnership on AI’s \*Responsible Practices for Synthetic Media\* (disclosure, consent, context).   
- \*\*Provenance:\*\* If available, attach \*\*Content Credentials (C2PA)\*\* via your editor; otherwise document the full toolchain here and in `README.md`.

## 8) Optional Video (not required for this submission)  
If you choose to add a talking-head later:  
- \*\*Wav2Lip:\*\* Lip-sync a still or face clip to `audio/final\_mix.mp3` via `inference.py` (checkpoint required).   
- \*\*SadTalker:\*\* Generate a talking head from a single image; configure `--preprocess`/`--size` per README.  
  
Document exact commands and checkpoints in `/tools-notes/` if you use these.

## 9) Files Produced  
- `audio/\*.wav` — individual speaker lines (Reporter/Experts)  
- `audio/final\_mix.mp3` — \*\*final deliverable\*\* (audio only)  
- `script/` — interview script and persona cards  
- `prompts/` — generation and voice style prompts  
- `process\_log/` — dated notes (planning, build, polish)  
- `PROCESS\_DOCUMENTATION\_Task\_06\_Deep\_Fake.md` (this document)

## 10) Submission  
- \*\*GitHub:\*\* Public repo titled `Task\_06\_Deep\_Fake` (track media with Git LFS if large).   
- \*\*Email:\*\* Send the repo link to \*\*jrstrome@syr.edu\*\*.   
- \*\*Time reporting:\*\* Complete Qualtrics check-in by \*\*Sept 1\*\*.

## 11) References (tooling & standards)  
- Piper TTS (GitHub README) — CLI, models, usage.   
- Piper Voices (Hugging Face) — `en\_US-amy-medium`, `en\_US-joe-medium`.   
- FFmpeg Concatenation (concat filter & demuxer).   
- Wav2Lip (GitHub) — inference examples.   
- SadTalker (GitHub) — inference and releases.   
- PAI Responsible Practices for Synthetic Media.   
- Adobe/CAI Content Credentials (C2PA) overview.  
  
\*See README “Sources” for scientific anchors (NIF, JET, AlphaFold 3, CRISPR therapy, perovskite–Si tandems).\*