

transcribe_audio.sh — Pre-Context

```
#!/usr/bin/env bash
set -euo pipefail

# -----
# transcribe-audio.sh
# Post-session transcription on Mac using whisper-cli
# Usage: transcribe-audio.sh [--de|--en|--auto] <audio.wav>
# -----

WHISPER="$HOME/Transkriptionen/whisper.cpp/build/bin/whisper-cli"
MODEL="$HOME/Transkriptionen/whisper.cpp/models/ggml-large-v3-turbo.bin"
VAD_MODEL="$HOME/Transkriptionen/whisper.cpp/models/ggml-silero-v6.2.0.bin"

# -----
# Argument parsing
# -----

LANG_MODE="en"
AUDIO=""

while [[ $# -gt 0 ]]; do
    case "$1" in
        --de)    LANG_MODE="de"; shift ;;
        --en)    LANG_MODE="en"; shift ;;
        --auto)  LANG_MODE="auto"; shift ;;
        -*)
            echo "Usage: transcribe-audio [--de|--en|--auto] <audio.wav>" >&2
            exit 2
            ;;
        *)
            AUDIO="$1"; shift ;;
    esac
done
```

```

done

if [[ -z "$AUDIO" ]]; then
    echo "  No audio file specified." >&2
    echo "    Usage: transcribe-audio [--de|--en|--auto] <audio.wav>" >&2
    exit 1
fi

if [[ ! -f "$AUDIO" ]]; then
    echo "  File not found: $AUDIO" >&2
    exit 1
fi

# -----
# Language options
# -----

case "$LANG_MODE" in
    en|de) LANG_OPTS=(-l "$LANG_MODE") ;;
    auto)  LANG_OPTS=() ;;
esac

# -----
# Output path: same dir as audio, .txt extension
# -----

AUDIO_DIR="$(dirname "$AUDIO")"
AUDIO_BASE="$(basename "$AUDIO" .wav)"
TRANSCRIPT_BASE="$AUDIO_DIR/${AUDIO_BASE}_transcript"

echo "  Audio:      $AUDIO"
echo "    Language:  $LANG_MODE"
echo "  Transcript: ${TRANSCRIPT_BASE}.txt"
echo "  ----"

# -----
# Run whisper-cli
# -----

    auto)  LANG_OPTS=() ;;
esac

```

```

# -----
# Output path: same dir as audio, .txt extension
# -----

AUDIO_DIR="$(dirname "$AUDIO")"
AUDIO_BASE="$(basename "$AUDIO" .wav)"
TRANSCRIPT_BASE="$AUDIO_DIR/${AUDIO_BASE}_transcript"

echo "Audio:      $AUDIO"
echo "Language:   $LANG_MODE"
echo "Transcript: ${TRANSCRIPT_BASE}.txt"
echo "----"

# -----
# Run whisper-cli
# -----

"$WHISPER" \
  -m "$MODEL" \
  "${LANG_OPTS[@]}" \
  --vad \
  -vm "$VAD_MODEL" \
  --output-txt \
  --output-srt \
  -of "$TRANSCRIPT_BASE" \
  -f "$AUDIO"

# Strip ANSI escape sequences and carriage returns from txt output
sed -i '' 's/\x1b\[ [0-9;]*[mGKH]//g; s/\r//g' "${TRANSCRIPT_BASE}.txt"

echo " Done: ${TRANSCRIPT_BASE}.txt"

```