

Speech to Text, Text to
Speech

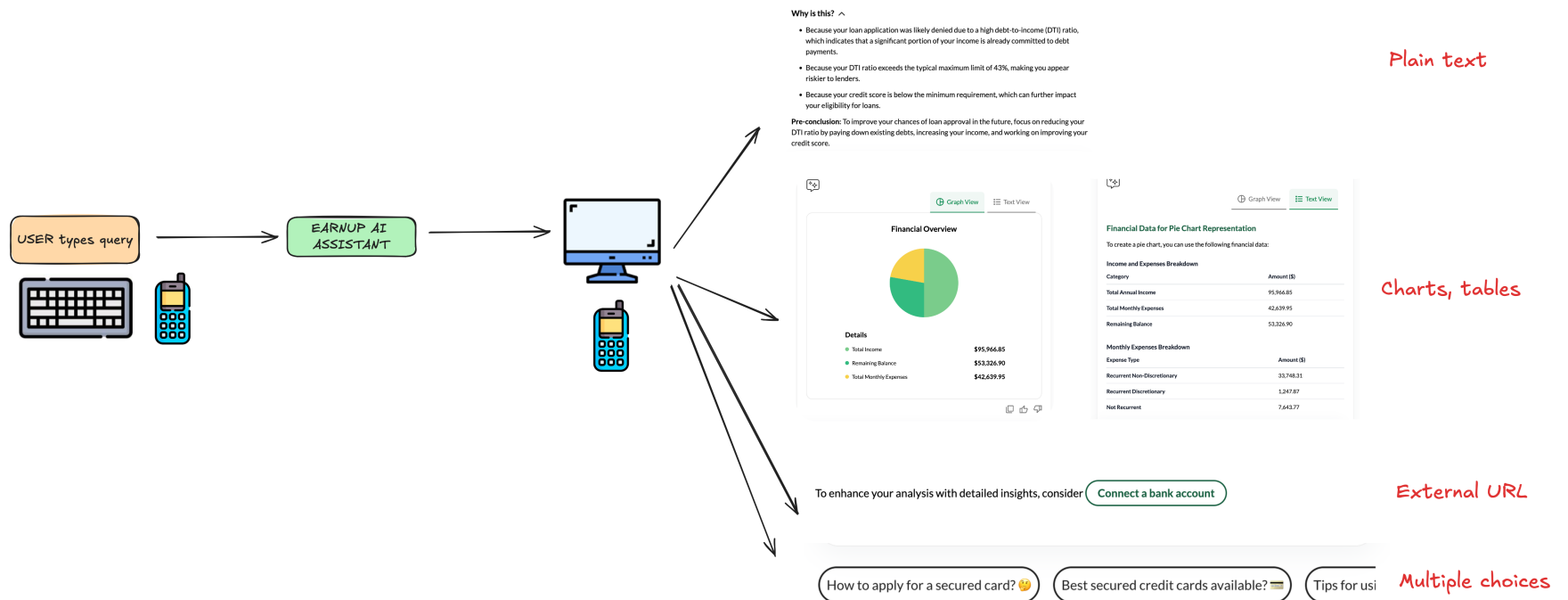
Challenges

- Artificial or Robotic-Sounding Speech
- Inaccurate Pronunciation
- Lack of Emotion or Expression
- Limited Language Support
- Technical Limitations
- Unnatural Pausing or Pacing
- Background Noise or Statics

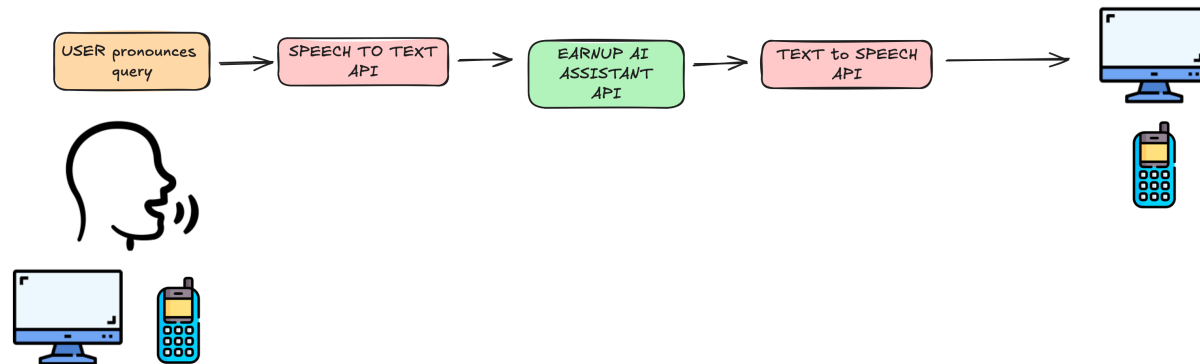
Building the best experiences

- HD Voice Technology
- Audio codecs
- Microphone and Speaker quality
- Noise cancellation
- Acoustic Modelling
- NLP
- Real Time audio processing
- Voice biometrics
- End to End encryption
- Different languages (Spanish)

Current mode





Introducing voice



Solutions

- [Deepgram](#)
- [Assembly AI](#)
- [Murf.ai](#)
- [Google Cloud AI Speech to Text service](#)
- Krisp: <https://krisp.ai/>
- [AWS transcribe](#)
- [Microsoft Bing Speech API](#)
- openAI Whisper

openAI vs. Deepgram

| |  Deepgram |  OpenAI Whisper <small>Fully managed by Deepgram</small> |
|--------------------------------------|-----------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| FEATURES AND CAPABILITIES | | |
| Batch process (1hr of audio) | ~30s | ~230s (large model) |
| Accuracy (WER) | 8.4 | 13.2 |
| Diarization (separate per speaker) | Up to 10 | Not available |
| Tailored speech models | ✓ | ✗ |
| Word level timestamps | ✓ | ✗ |
| Deep Search (audio) | ✓ | ✗ |
| Paragraphs | ✓ | ✗ |
| Custom Vocabulary (keyword boosting) | ✓ | ✗ |
| Redaction | ✓ | ✗ |
| Summarization | ✓ | ✗ |
| Punctuation | ✓ | ✓ |
| Profanity Filter | ✓ | ✓ |
| Numeral Formatting | ✓ | ✓ |
| PRICING | | |
| Pre-recorded per minute | Starting at \$0.0043 | Starting at \$0.0060 |
| Streaming per minute | Starting at \$0.0059 | ✗ |

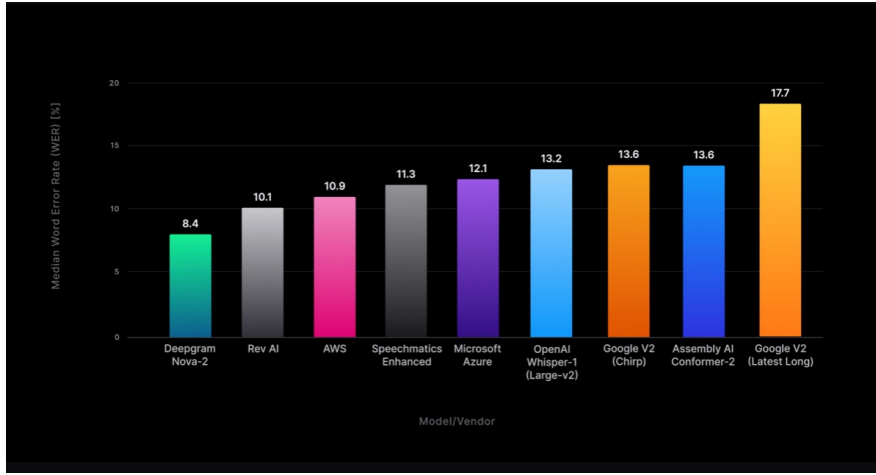
| Model | Usage |
|---------|--------------------------|
| Whisper | \$0.006 / minute |
| TTS | \$15.000 / 1M characters |
| TTS HD | \$30.000 / 1M characters |

- openAI could be a good option for a one stop shop, for transcription (Sound to Text) and audio output (Text to Speech), in both english and Spanish
- Already using openAI services
- 6 voices options.
- No external API (Deepgram).
- Need to check translation accuracy, latency.
- Need to check data retention.

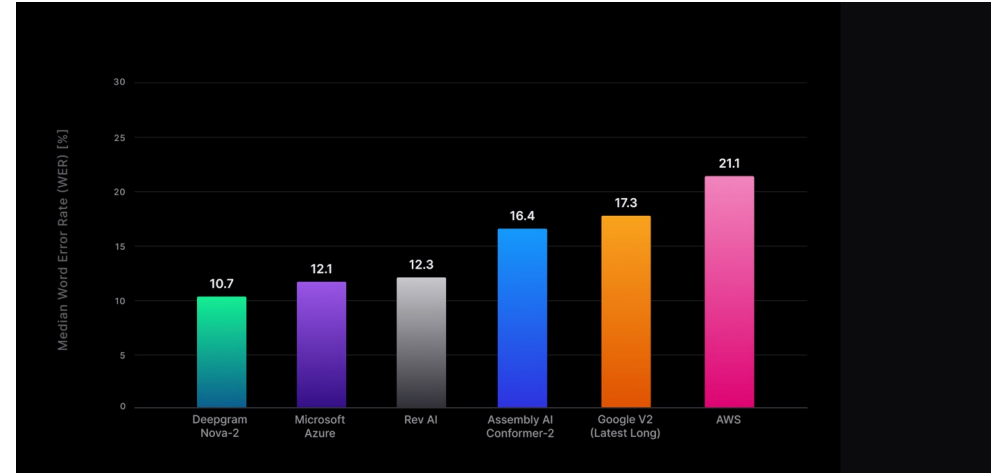
Solutions

- OpenAI + Assembly.ai + Elevenlabs
 - 3 different api's
 - Assembly.ai:
 - STT, audio intelligence ; 92.5% accuracy ; 30ms latency ; diarization ; international language support
 - <https://www.assemblyai.com>
 - Eleven Labs:
 - TTS, voice generator, text sound effect(animal), voice cloning
- OpenAI + Deepgram
 - 2 different api's
 - <300ms latency (including network latency)
 - Language detection, multilingual switch
 - Soc2, HIPPA, GDPR, CCPA, PCA compliants
 - Encryption of any sensitive or confidential information (including PHI and PCI data) in transit and at rest

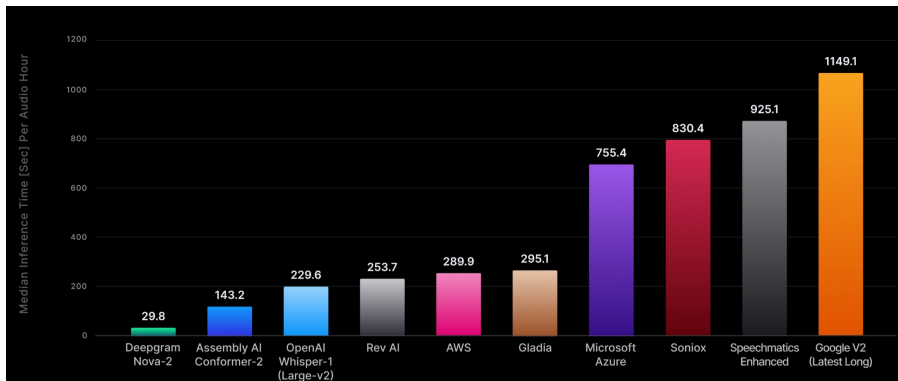
WER batch



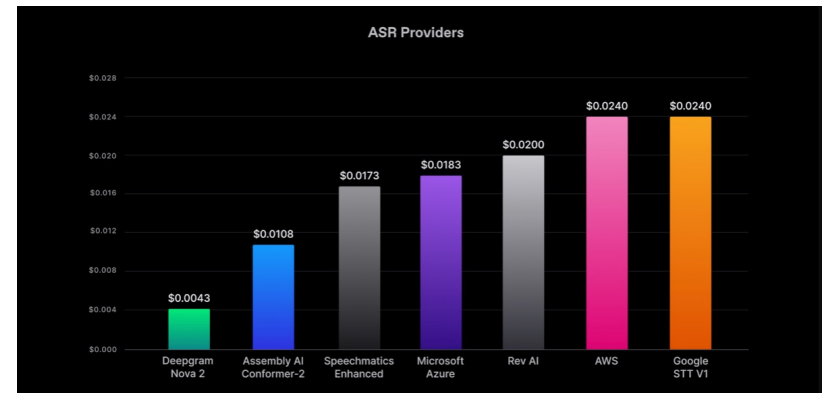
WER real time



Time to transcribe 1h batch audio



Pricing



```
DEEPGRAM_URL = f"https://api.deepgram.com/v1/speak?model={self.MODEL_NAME}&encoding=linear16&sample_rate=24000"
headers = {
    "Authorization": f"Token {self.DG_API_KEY}",
    "Content-Type": "application/json"
}
payload = {
    "text": text
}
```

- Encoding: expected encoding of submitted audio.
- sample_rate: refers to the number of samples of audio carried per second, measured in Hertz (Hz).
- Choosing the appropriate sample rate is crucial as it directly impacts the audio quality and file size of the output. Higher sample rates typically result in better audio quality but may increase the file size, while lower sample rates may reduce file size but can compromise audio fidelity.
- Language: <https://developers.deepgram.com/docs/models-languages-overview>

Deepgram supports the following audio coding algorithms:

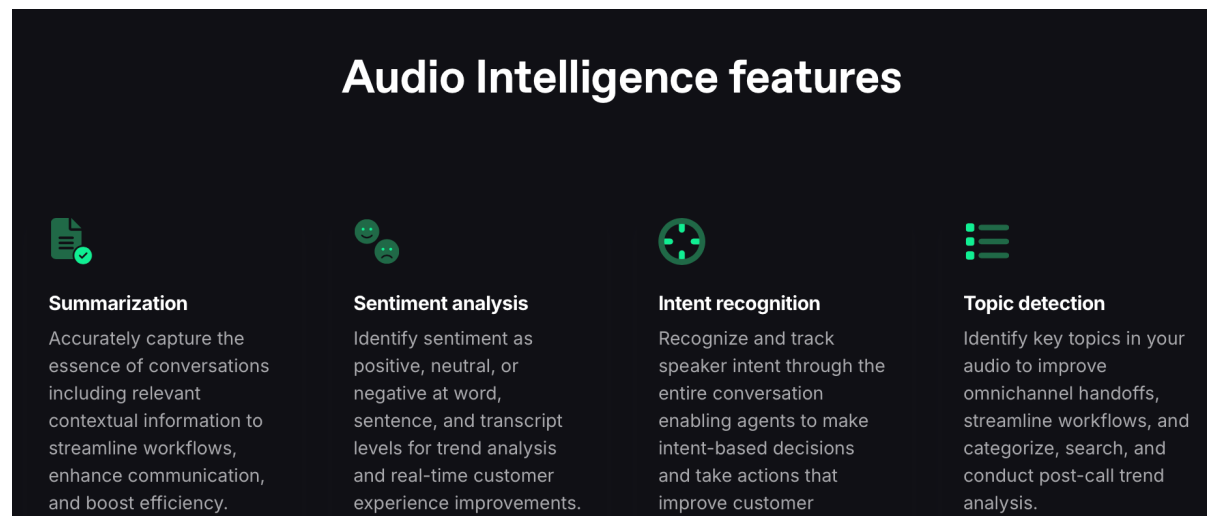
- linear16 : 16-bit, little endian, signed PCM WAV data
- flac : Free Lossless Audio Codec (FLAC) encoded data
- mulaw : Mu-law encoded WAV data
- amr-nb : Adaptive Multi-Rate (AMR) narrowband codec
- amr-wb : Adaptive Multi-Rate (AMR) wideband codec
- opus : Ogg Opus
- speex : Speex
- g729 : G729 low-bandwidth (required for both raw and containerized audio)

| Encoding | Container | Sample Rate (Hz) | Bitrate (bps) |
|---------------|------------------------|-----------------------------------------------------------|--------------------------------------------------|
| linear16 | wav (default), none | 8000 , 16000 , 24000 (default), 32000 , 48000 | Not Applicable |
| mulaw | wav (default), none | 8000 (default), 16000 | Not Applicable |
| alaw | wav (default), none | 8000 (default), 16000 | Not Applicable |
| mp3 (default) | Not Applicable | Not Configurable (set to 22050) | 32000 , 48000 (default) |
| opus | ogg (default) | Not Configurable (set to 48000) | Default: 12000 Range: >= 4000 and <= 65000 |
| flac | Not Applicable | 8000 , 16000 , 22050 , 32000 , 48000 | Not Applicable |
| aac | Not Applicable | Not Configurable (set to 22050) | Default: 48000 , Range: >= 4000 and <= 192000 |

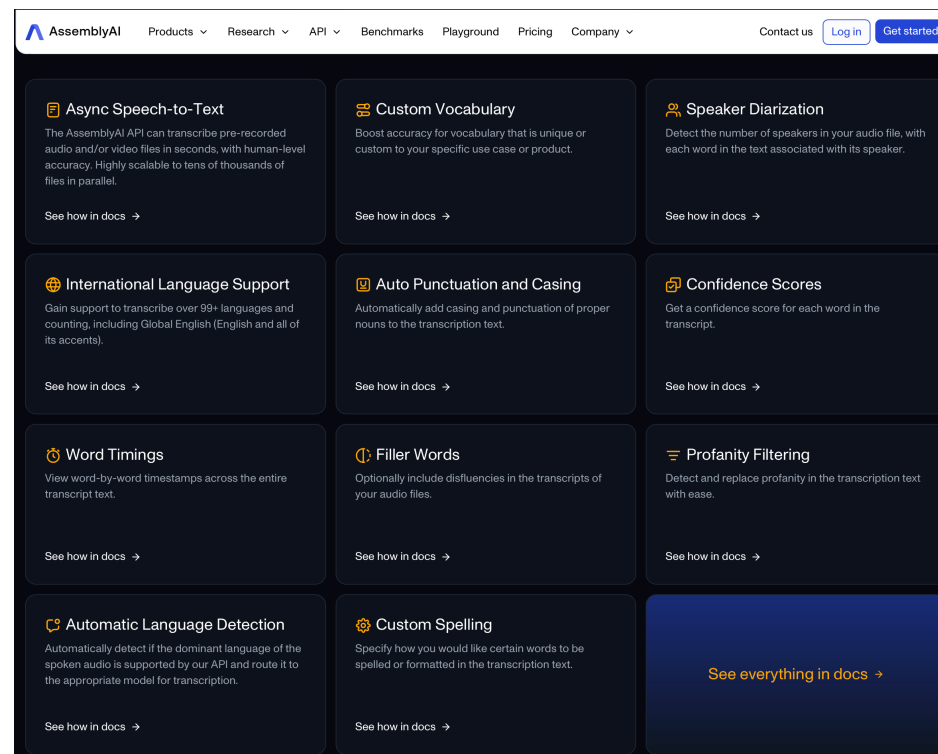
Options for transcription (nova-2 model)

```
options = LiveOptions(  
    model="nova-2",  
    punctuate=True,  
    language="en-US",  
    encoding="linear16",  
    channels=1,  
    sample_rate=16000,  
    endpointing=True  
)
```

Deepgram intelligence features



Assembly.ai features



EleveLabs free tier

- 10 minutes of ultra-high quality text to speech per month
- Generate speech in 32 languages using thousands of unique voices
- Translate content with automatic dubbing
- Create custom, synthetic voices
- Generate sound effects
- API access
- **Standard models** (Multilingual v2, Multilingual v1, English v1) are optimized for quality and accuracy, ideal for content creation. These models offer the best quality and stability but have higher latency.
- **Turbo models** (Turbo v2, Turbo v2.5) are designed for low-latency applications like real-time conversational AI. They deliver great performance with faster processing speeds, though with a slight trade-off in accuracy and stability.
- Security:
 - End to end encryption
 - AICPA, SOC2
 - GDPR compliant
 - No-retention mode

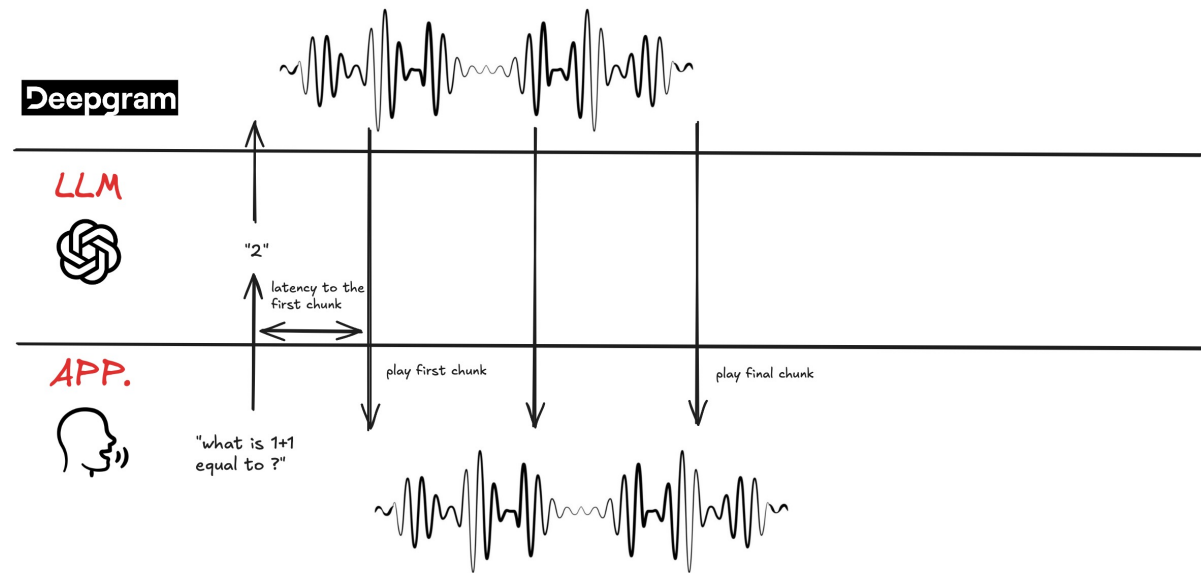
Deepgram

Voice Feature mode



TTS: deepgram streaming

Text to Speech: Deepgram Streaming



Considerations

- Filler words to disguise latency
- Interruptions
 - More difficult to handle
 - Software engineering improvement (interrupting / resuming an audio stream) rather than an AI improvement
- Start the response before the user finishes:
 - Human starts forming their answer before the other person is done talking
 1. Stream the speech to the LLM
 2. Have the LLM predict the rest of the user speech
 3. Form response based on expected user speech