Comparative analysis of STT models

1. Introduction

Speech-to-Text (STT) technology converts spoken language into written text, enabling voice-based applications to interact with users through natural speech. The technology works by recording audio, breaking it into segments, and using machine learning algorithms to match sounds with text-based words. Sound is transformed into numbers by capturing its waveform and extracting acoustic features like frequency, amplitude, timbre, and other characteristics, which are represented as spectrograms or coefficients for machine analysis. This process allows the system to "translate" speech into text for further use [1].

In real-time applications like Ale, a virtual assistant, the key challenges for STT models are low latency and high accuracy. Real-time transcription ensures smooth user experience, while accuracy is crucial for capturing the user's intent without errors.

This analysis evaluates two popular STT models:

OpenAI Whisper, a model designed for high accuracy and multilingual support, making it a versatile solution for diverse language data [2].

Deepgram, on the other hand, is a platform optimized for high-speed performance and scalability, particularly suited for real-time applications with large data volumes [3].

2. Model Overview

OpenAI Whisper: Whisper is an open-source transformer-based model trained on 680,000 hours of data, offering robustness to accents, noise, and diverse language scenarios (supports around 100 languages). It is free to use but requires significant resources for customization and deployment, making it less optimal for real-time applications.

Deepgram: Deepgram is a commercial platform optimized for high speed and low latency, making it ideal for real-time applications. It supports 30 languages, provides a flexible API for integration and customization, but requires a subscription for use.

Both models operate on a similar principle: acoustic features, such as frequency characteristics, are extracted from the audio signal using spectral transformation. This data is converted into a spectrogram, which is then fed into a neural network. Whisper uses a transformer architecture to map acoustic features to text, ensuring high accuracy and multilingual support. Deepgram also relies on deep learning techniques but is optimized for real-time processing, delivering high speed and low latency.

3. Comparison criteria

3.1. Accuracy

Word Error Rate (WER): Word Error Rate (WER) is a key metric for evaluating the accuracy of speech recognition systems. It is calculated as

$$WER = (S+D+I)/N, [4]$$

where S is the number of substitutions, D is deletions, I is insertions, and N is the total number of words in the reference text.

For OpenAI Whisper, the WER ranges from 15% for English to 19% for Spanish, demonstrating robust performance across multiple languages despite slight declines in noisy conditions [5].

Deepgram demonstrates a relative WER reduction of approximately 36% compared to Whisper, resulting in an estimated WER between 10% and 12%, depending on the testing conditions. It is optimized for real-time applications and shows improved accuracy over Whisper in similar scenarios [6].

Handling Accents and Dialects: Whisper excels at handling diverse accents and dialects, thanks to its multilingual training data, which includes a wide variety of linguistic and acoustic patterns. However, Whisper may show reduced accuracy when processing heavy regional accents, such as Indian or South African English, due to the model's exposure to a limited range of these accents during training. Recent improvements to Whisper are focused on increasing its robustness to these diverse accents [7]. Deepgram also supports accents and dialects effectively, but its performance may vary depending on the language and dataset used. Deepgram allows customization through its API, enabling fine-tuning for specific regional accents, which can significantly improve accuracy. In practice, users have reported that while Deepgram handles standard English accents well, more uncommon accents, like certain Asian or African dialects, require additional configuration to achieve optimal performance [8].

3.2. **Speed and Latency**

OpenAI Whisper: The Whisper model, especially in its larger variants, delivers high accuracy but requires significant computational resources, resulting in increased inference times. According to available data, processing one hour of audio can take up to 229.6 seconds [6].

Deepgram: Deepgram is optimized for fast and efficient real-time speech recognition. Processing one hour of audio takes approximately 29.8 seconds, making it about five times faster than Whisper [6].

The actual performance of both models may vary depending on the hardware, complexity of the audio material, and specific usage scenarios.

3.3. Ease of Integration

Whisper:

- Whisper is an open-source model available through platforms like Hugging Face and other repositories, making it straightforward to deploy locally and use. Its open nature allows developers to modify and adapt the model for specific tasks.
- A key advantage is the ability to run it on local servers without relying on cloud services, which is ideal for scenarios with high-security requirements [9].

Deepgram:

- Deepgram offers a cloud-based solution with models accessible via API [3]. This simplifies integration into existing workflows, especially for Python-based applications and other programming environments.
- The platform provides SDKs for various programming languages, making it attractive for developers aiming to quickly integrate ASR into their projects.

3.4. Resource Requirements

Whisper: Whisper requires significant computational resources for on-premise deployment. GPUs or high-performance CPUs are essential for efficient operation, especially when using larger model variants. Memory capacity is also important, as Whisper models can be quite large [10].

Deepgram: Deepgram operates in the cloud, minimizing resource usage on the client side. This means developers don't need powerful hardware. However, performance relies on the quality of the internet connection, as processing is done remotely, which may introduce latency [11].

3.5. Cost and Scalability

Whisper: Whisper is free and open-source, making it appealing for smaller projects with limited budgets. However, deploying it requires dedicated infrastructure, including powerful hardware and maintenance costs, which can increase the overall expense for large-scale projects [2].

Deepgram: Deepgram operates on a subscription-based pricing model, making it convenient for quick deployment. However, costs can rise significantly with high-volume real-time transcription, particularly for scalable enterprise solutions. That said, the provided support and updates may offset these expenses [3].

4. Use Case in Ale

Integration of Models:

- **Whisper:** Suitable for scenarios requiring robust transcription on diverse datasets or offline processing. It provides high accuracy and flexibility but demands significant computational resources.
- **Deepgram:** Ideal for tasks with minimal setup that require real-time transcription. Its cloud-based architecture allows seamless integration into Ale's existing audio pipeline.

Recommendation:
For Ale, **Deepgram** is the preferred model if the priority is low latency, scalability, and high performance in real-time applications. However, if autonomy and support for complex multilingual scenarios are critical, Whisper would be a better choice. The decision depends on whether Ale values real-time performance or versatility more.

5. References and Benchmarks

[1] He, Y., & Wu, H. (2019). *nnAudio: An on-the-fly GPU audio to spectrogram conversion toolbox using 1D convolution neural networks*. arXiv. Retrieved December 13, 2024, from https://arxiv.org/abs/1912.12055

[2] OpenAI. (2021). *Whisper: Open-source automatic speech recognition model*. GitHub. Retrieved December 13, 2024, from https://github.com/openai/whisper

[3] Deepgram. (n.d.). *Deepgram API documentation*. Retrieved December 13, 2024, from https://developers.deepgram.com/docs

[4] Wikipedia contributors. (n.d.). *Word error rate*. Wikipedia, The Free Encyclopedia. Retrieved December 13, 2024, from https://en.wikipedia.org/wiki/Word_error_rate

[5] Heikinheimo, H. (2023, April 4). *Analyzing OpenAI's Whisper ASR models: Word error rates across languages*. Speechly. Retrieved December 13, 2024, from https://www.speechly.com/blog/analyzing-open-ais-whisper-asr-models-word-error-rates-across-languages

[6] Deepgram. (n.d.). *OpenAI vs. Deepgram: Speech recognition comparison*. Retrieved December 13, 2024, from https://deepgram.com/compare/openai-vs-deepgram-alternative

[7] Graham, C., & Roll, N. (2024). Evaluating OpenAI's Whisper ASR: Performance analysis across diverse accents and speaker traits. *JASA Express Letters*, *4*(2), 025206. https://doi.org/10.1121/10.0024876

[8] Lam, K. (2022, March 29). *Deepgram now offers more than 20 language and dialect speech models*. Deepgram. Updated June 13, 2024. Retrieved December 13, 2024, from https://deepgram.com/learn/deepgram-language-speech-models

[9] Hugging Face. (n.d.). *Whisper: Open-source automatic speech recognition model*. Retrieved December 13, 2024, from https://huggingface.co/docs/transformers/v4.46.2/en/model_doc/whisper

[10] OpenAI. (2022). *Whisper: Discussion on computational requirements for large models*. GitHub. Retrieved December 13, 2024, from https://github.com/openai/whisper/discussions/5

[11] Deepgram. (n.d.). *Self-hosted solutions*. Retrieved December 13, 2024, from https://deepgram.com/self-hosted