**Abstract**  
Speech-to-text (STT) models are essential for applications like voice assistants, transcription services, and automated form-filling systems. This study evaluates STT models based on accuracy, computational efficiency, multilingual support, latency, and integration capabilities. The models analyzed include OpenAI Whisper, DeepSpeech, Wav2Vec 2.0, and Google Speech-to-Text API. This research presents a comparative analysis of these models, highlighting their advantages, limitations, and best use cases.

**1. Introduction**

Speech recognition technology enables hands-free interaction, making it invaluable for form-filling applications. Various STT models employ different architectures and training methodologies, affecting their accuracy and performance. This paper compares key models based on core performance metrics to determine the best fit for voice-based form automation.

**2. Methodology**

We analyze four widely used STT models:

* **OpenAI Whisper**
* **DeepSpeech (by Mozilla)**
* **Wav2Vec 2.0 (by Facebook AI)**
* **Google Speech-to-Text API**

**Evaluation Criteria:**

* **Accuracy (Word Error Rate - WER)**
* **Latency (Processing time per second of audio)**
* **Multilingual Support**
* **Hardware Requirements**
* **Ease of Integration**

**3. Model Analysis**

**3.1 OpenAI Whisper**

**Pros:**  
✔ High multilingual support (99+ languages)  
✔ Robust in noisy environments  
✔ Open-source and fine-tunable  
✔ Handles diverse accents well

**Cons:**  
✖ High computational demand (GPU recommended)  
✖ Higher latency than cloud-based alternatives

**Best Use Case:** Suitable for applications needing offline, multilingual, and high-accuracy transcription.

**3.2 DeepSpeech (Mozilla)**

**Pros:**  
✔ Lightweight and efficient for on-device use  
✔ Open-source and customizable  
✔ No dependency on cloud services

**Cons:**  
✖ Lower accuracy than deep-learning-based models  
✖ Limited multilingual support  
✖ Requires model training for better accuracy

**Best Use Case:** Ideal for applications that require offline speech recognition with minimal hardware.

**3.3 Wav2Vec 2.0 (Facebook AI)**

**Pros:**  
✔ State-of-the-art accuracy with self-supervised learning  
✔ Can be fine-tuned with small datasets  
✔ Works well with low-quality audio

**Cons:**  
✖ Requires GPU for efficient processing  
✖ Limited built-in multilingual support

**Best Use Case:** Best for high-accuracy transcription with adaptable training on domain-specific datasets.

**3.4 Google Speech-to-Text API**

**Pros:**  
✔ Real-time processing with low latency  
✔ Supports 125+ languages  
✔ Easy integration with cloud applications  
✔ High accuracy with contextual adaptation

**Cons:**  
✖ Paid service (cost increases with usage)  
✖ Requires internet connectivity  
✖ Privacy concerns with cloud processing

**Best Use Case:** Best for real-time, scalable, and cloud-based applications requiring high-speed speech recognition.

**4. Comparative Analysis**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **Accuracy (WER)** | **Latency** | **Multilingual Support** | **Hardware Requirements** | **Integration Ease** |
| **OpenAI Whisper** | ⭐⭐⭐⭐⭐ (Low WER) | High | Yes (99+ languages) | High (GPU recommended) | Medium |
| **DeepSpeech** | ⭐⭐⭐ (Medium WER) | Low | No | Low (CPU sufficient) | High |
| **Wav2Vec 2.0** | ⭐⭐⭐⭐⭐ (Low WER) | Medium | Limited | High (GPU required) | Medium |
| **Google STT API** | ⭐⭐⭐⭐ (Low WER) | Low | Yes (125+ languages) | Cloud-based | Easy |

**5. Discussion**

* **Best for High Accuracy:** OpenAI Whisper and Wav2Vec 2.0.
* **Best for Real-Time Processing:** Google Speech-to-Text API.
* **Best for On-Device Use:** DeepSpeech.
* **Best for Multilingual Support:** OpenAI Whisper and Google STT API.
* **Best for Cost-Effectiveness:** DeepSpeech (open-source, runs on local hardware).

**6. Conclusion**

There is no one-size-fits-all STT model. The choice depends on the application's needs:

* **For high-accuracy and multilingual transcription**, OpenAI Whisper is the best choice.
* **For real-time processing**, Google Speech-to-Text API offers the lowest latency.
* **For on-device processing with lower hardware requirements**, DeepSpeech is the best option.
* **For self-supervised training on custom datasets**, Wav2Vec 2.0 is ideal.

For **voice-based form filling**, a **hybrid approach** is recommended:

* **Use OpenAI Whisper** for offline processing and multilingual support.
* **Use Google STT API** for real-time, cloud-based transcription.

**7. References**

* OpenAI Whisper: [Whisper GitHub](https://github.com/openai/whisper)
* DeepSpeech: [DeepSpeech GitHub](https://github.com/mozilla/DeepSpeech)
* Wav2Vec 2.0: [Facebook AI Wav2Vec 2.0](https://ai.facebook.com/blog/wav2vec-2-0/)
* Google STT API: [Google Cloud Speech-to-Text](https://cloud.google.com/speech-to-text)

This comparative study provides insights into STT model selection, aiding in the development of efficient voice-based form-filling applications.