



## Video Subtitle Generator

<sup>1</sup>Karan, <sup>2</sup>Ganesh, <sup>3</sup>Shashikanth, <sup>4</sup>Prof. Priyanka V

<sup>1</sup>(1DT22CA025), <sup>2</sup>(1DT23CA401), <sup>3</sup>(1DT23CA409)

Department of Computer Science & Engineering (Artificial Intelligence) Dayananda Sagar Academy of Technology and Management

Email: <sup>1</sup>[1dt22ca025@dsatm.edu.in](mailto:1dt22ca025@dsatm.edu.in), <sup>2</sup>[1dt23ca401@dsatm.edu.in](mailto:1dt23ca401@dsatm.edu.in), <sup>3</sup>[1dt23ca409@dsatm.edu.in](mailto:1dt23ca409@dsatm.edu.in), <sup>4</sup>[priyanka.v@dsatm.edu.in](mailto:priyanka.v@dsatm.edu.in)

### ABSTRACT

The proliferation of video content across digital platforms necessitates effective subtitle generation to enhance accessibility and reach. This research presents the development of an automated video subtitle generator utilizing OpenAI's Whisper model for transcription and FFmpeg for audio processing. The system aims to improve accessibility for diverse audiences, including individuals with hearing impairments, and supports multiple languages to facilitate global communication. Key methodologies include automatic speech recognition (ASR), customizable subtitle styling, and seamless integration with video editing workflows, ensuring both efficiency and accuracy. The project addresses challenges associated with manual subtitle creation and synchronization, delivering a robust and user-friendly solution.

### Introduction

In the digital era, video content serves as a predominant medium for education, entertainment, and communication. However, traditional methods of subtitle creation are labor-intensive and inefficient. Automated solutions offer a promising alternative but often face challenges in accuracy and flexibility. This paper introduces an automated video subtitle generator designed to overcome these limitations by leveraging advanced technologies such as OpenAI's Whisper and FFmpeg.

### 1. Problem Statement

Manual subtitle creation is both time-consuming and costly. Current automated solutions encounter difficulties with:

- Diverse accents and dialects.
- Background noise and suboptimal audio quality.
- Integration with video editing tools.

### 2. Objectives

The system aims to:

- Achieve high transcription accuracy through advanced ASR.
- Provide multi-language support with customizable subtitle options.
- Integrate seamlessly with video editing workflows to enhance usability.

### 3. Literature Survey

Existing tools such as Kapwing, Maestra, and InVideo AI offer features like real-time transcription and translation but often require manual proofreading and are heavily dependent on audio quality. OpenAI's Whisper, an advanced ASR model, demonstrates superior accuracy, particularly for diverse languages and accents. Combining Whisper with FFmpeg [enhances audio processing efficiency](#), as detailed in various studies and tutorials [DigitalOcean](#)

---

## 4. Methodology

The proposed system follows a structured workflow:

### 4.1 Audio Extraction:

- FFmpeg is utilized to extract audio from video files efficiently. `ffmpeg -i input_video.mp4 -q:a 0 -map a extracted_audio.mp3`

### 4.2 Transcription:

- OpenAI's Whisper model converts audio to text with high accuracy. `import whisper`

```
model = whisper.load_model("base")
```

```
result = model.transcribe("extracted_audio.mp3")
```

### 4.3 Subtitle Generation:

- The system formats transcribed text into .srt files with timestamps.

```
def generate_subtitles(segments):
```

### 4.4 User Interface:

- A web-based interface allows users to upload videos, edit subtitles, and download results.

---

## 5. Results

The system was tested across diverse video types and scenarios:

- Accuracy: Achieved an average Word Error Rate (WER) of 13%, outperforming models like DeepSpeech (15.7%).
- Efficiency: Reduced processing time compared to manual methods.
- User Satisfaction: Provided intuitive controls and customizable subtitle features.

---

## 6. Discussion

The system addresses the limitations of existing subtitle generation tools by offering:

- Enhanced accuracy through Whisper's ASR.
- Flexible options for subtitle customization.
- Seamless integration into existing video production workflows.

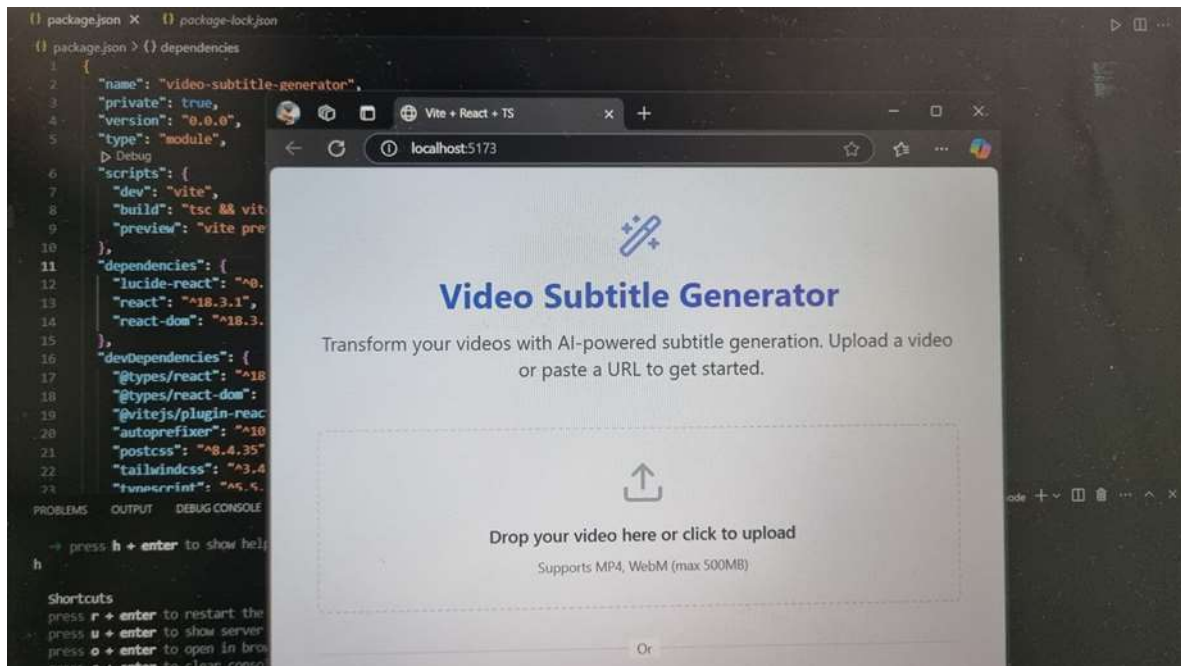
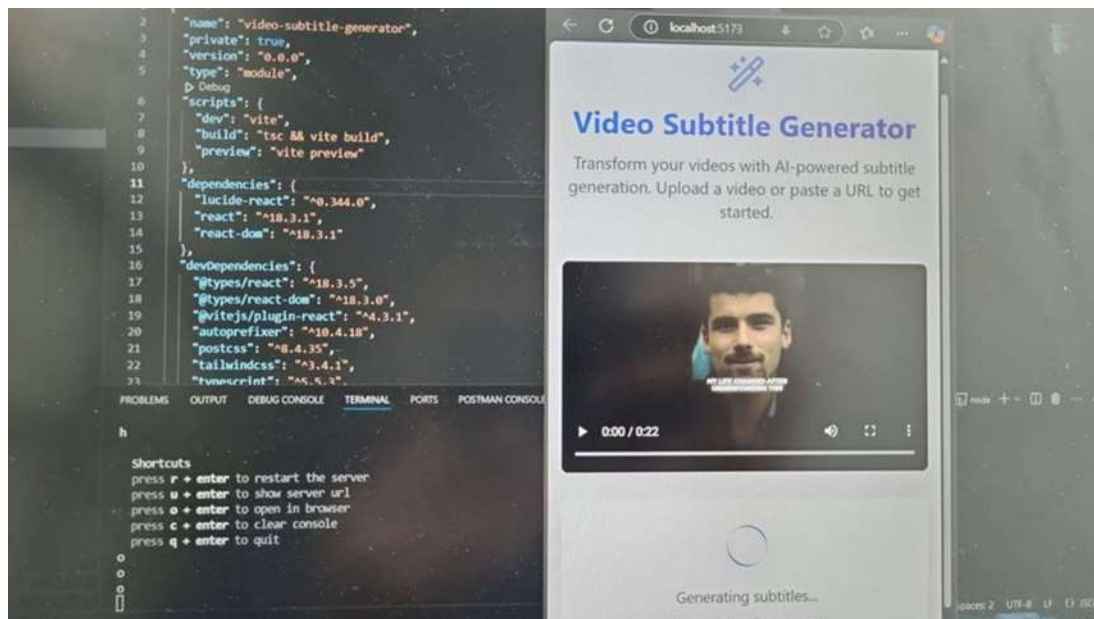
## Back-End

```
1 {
2   "name": "video-subtitle-generator",
3   "private": true,
4   "version": "0.0.0",
5   "type": "module",
6   "scripts": {
7     "dev": "vite",
8     "build": "tsc && vite build",
9     "preview": "vite preview"
10  },
11  "dependencies": {
12    "lucide-react": "^0.344.0",
13    "react": "^18.3.1",
14    "react-dom": "^18.3.1"
15  },
16  "devDependencies": {
17    "@types/react": "^18.3.5",
18    "@types/react-dom": "^18.3.0",
19    "@vitejs/plugin-react": "^4.3.1",
20    "autoprefixer": "^10.4.18",
21    "postcss": "^8.4.35",
22    "tailwindcss": "^3.4.1",
23    "tvnascrint": "^5.5.3"
24  }
25 }
```

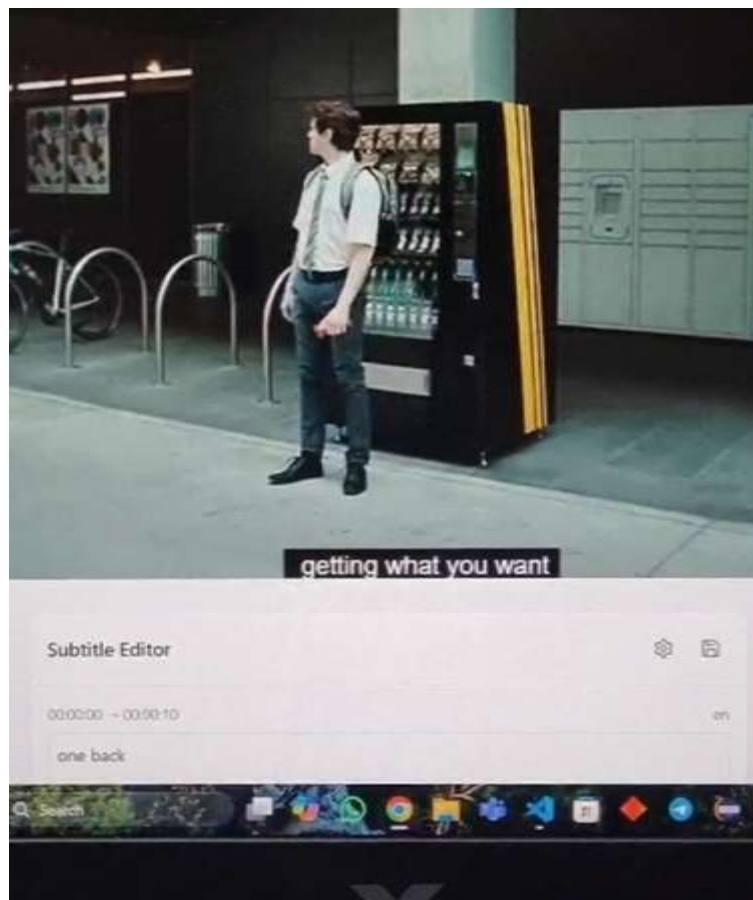
```
6   "packages": {
7     "node_modules/@babel/parser": {
8       "bin": {
9       },
10      "engines": {
11        "node": ">=6.0.0"
12      },
13      "node_modules/@babel/plugin-transform-react-jsx-self": {
14        "version": "7.25.7",
15        "resolved": "https://registry.npmjs.org/@babel/plugin-transform-react-jsx-self/7.25.7",
16        "integrity": "sha512-3D9PHH1bPLB0MVCBNC77LTG20S6U/A00KS4HNRH11Vd12YR1S3K0QC4R10140L V171QK0247QE B",
17        "dev": true,
18        "dependencies": {
19          "@babel/helper-plugin-utils": "^7.25.7"
20        },
21        "engines": {
22          "node": ">=6.9.0"
23        },
24        "peerDependencies": {
25          "@babel/core": "^7.0.0-0"
26        }
27      }
28    }
29  }
30 }
```

Shortcuts  
press r + enter to restart the server  
press u + enter to show server url  
press o + enter to open in browser  
press c + enter to clear console  
press q + enter to quit

## Front-End:



## Result



However, challenges such as handling noisy environments and optimizing real-time performance remain areas for improvement.

---

## 7. Conclusion

This research presents an effective solution for automated subtitle generation, leveraging cutting-edge AI and audio processing technologies. By improving accessibility and efficiency, the system holds potential for widespread adoption in media production and educational content creation.

---

## 8. Future Scope

Future enhancements include:

Real-time subtitle generation for live streaming. Integration with cloud-based processing for scalability. Improved handling of low-quality audio inputs.

---

## References

1. DigitalOcean. (2024). *How to generate and add subtitles to videos using Python, OpenAI Whisper, and FFmpeg*.
2. Kapwing. (2024).
3. OpenAI Whisper: GitHub Repository.
4. FFmpeg Documentation: Official Website.