Project Report

Zero AI Speech-to-Text Transcription

Team ID - DARPG_45

Problem Statement - 3

Team Members -

1. ATHRVA DESHMUKH (https://github.com/athrvadeshmukh)
2. UJJWAL GUPTA (https://github.com/masterujjval)
3. SONU KUSHWAHA (https://github.com/Sonu7804)

Submit To -

Online Hackathon on Data-driven Innovation for Citizen Grievance Redressal organized by the Department of Administrative Reforms & Public Grievances (DARPG) of the Ministry of Personnel, Public Grievances & Pensions.

PROBLEM STATEMENT 3 : Evaluate and optimize an existing open-source speech-to-text transcription tool for accurately converting feedback calls related to citizen grievances into English text. The goal is to benchmark the tool's performance and implement enhancements to achieve measurable improvements in transcription accuracy for calls in Hindi, English, and Hinglish. This project does not involve creating a new system but rather focuses on refining an already established open-source solution.

Project Link -

https://github.com/athrvadeshmukh/DARPG-HACKATHON.git

Executive Summary

The project aims to evaluate and optimize an existing open-source speech-to-text transcription tool for accurately converting feedback calls related to citizen grievances into English text. The tool under consideration is ZeroAI.py, which utilizes the Whisper library for transcription tasks. The primary objective is to benchmark the tool's performance and implement enhancements to achieve measurable improvements in transcription accuracy for calls in Hindi, English, and Hinglish. The project does not entail creating a new system but rather focuses on refining an already established open-source solution.

Introduction

The objective of this project is to evaluate and optimize the open-source speech-to-text transcription tool, Whisper, for accurately converting feedback calls related to citizen grievances into English text. The project aims to benchmark the tool’s performance and implement enhancements to achieve measurable improvements in transcription accuracy for calls in Hindi, English, and Hinglish. Rather than creating a new system, the focus is on refining the already established open-source solution provided by Whisper. The proliferation of citizen feedback mechanisms necessitates efficient handling and processing of various communication channels, including voice calls. In contexts where feedback is provided in multilingual formats such as Hindi, English, and Hinglish, automated transcription tools play a crucial role in extracting actionable insights from diverse data sources. This project addresses the optimization of an existing speech-to-text transcription tool to enhance its accuracy and usability in handling feedback calls related to citizen grievances.

Objectives

- Evaluate the current performance of the open-source speech-to-text transcription tool.
- Identify key areas for improvement in transcription accuracy, particularly for calls in Hindi, English, and Hinglish.
- Implement enhancements and optimizations to the existing tool to achieve measurable improvements.
- Benchmark the performance of the optimized tool against the baseline.
- Provide recommendations for future enhancements and usage scenarios.

Whisper Overview

Whisper is a general-purpose speech recognition model developed by OpenAI. It is a Transformer sequence-to-sequence model trained on a large dataset of diverse audio. The model is designed to perform multilingual speech recognition, speech translation, spoken language identification, and voice activity detection.

Approach

Whisper utilizes a Transformer sequence-to-sequence model trained on various speech processing tasks. These tasks are jointly represented as a sequence of tokens to be predicted by the decoder, enabling a single model to replace many stages of a traditional speech-processing pipeline. The multitask training format employs special tokens that serve as task specifiers or classification targets.

Project Setup

Software Requirements

- Python 3.8-3.11
- PyTorch 1.10.1
- ffmpeg
- rust (if required)

Installation

The Whisper package can be installed via pip using the following commands:
Bash -
pip install -U openai-whisper

Windows -
pip install git+https://github.com/openai/whisper.git
pip install ffmpeg-python
pip install pydub

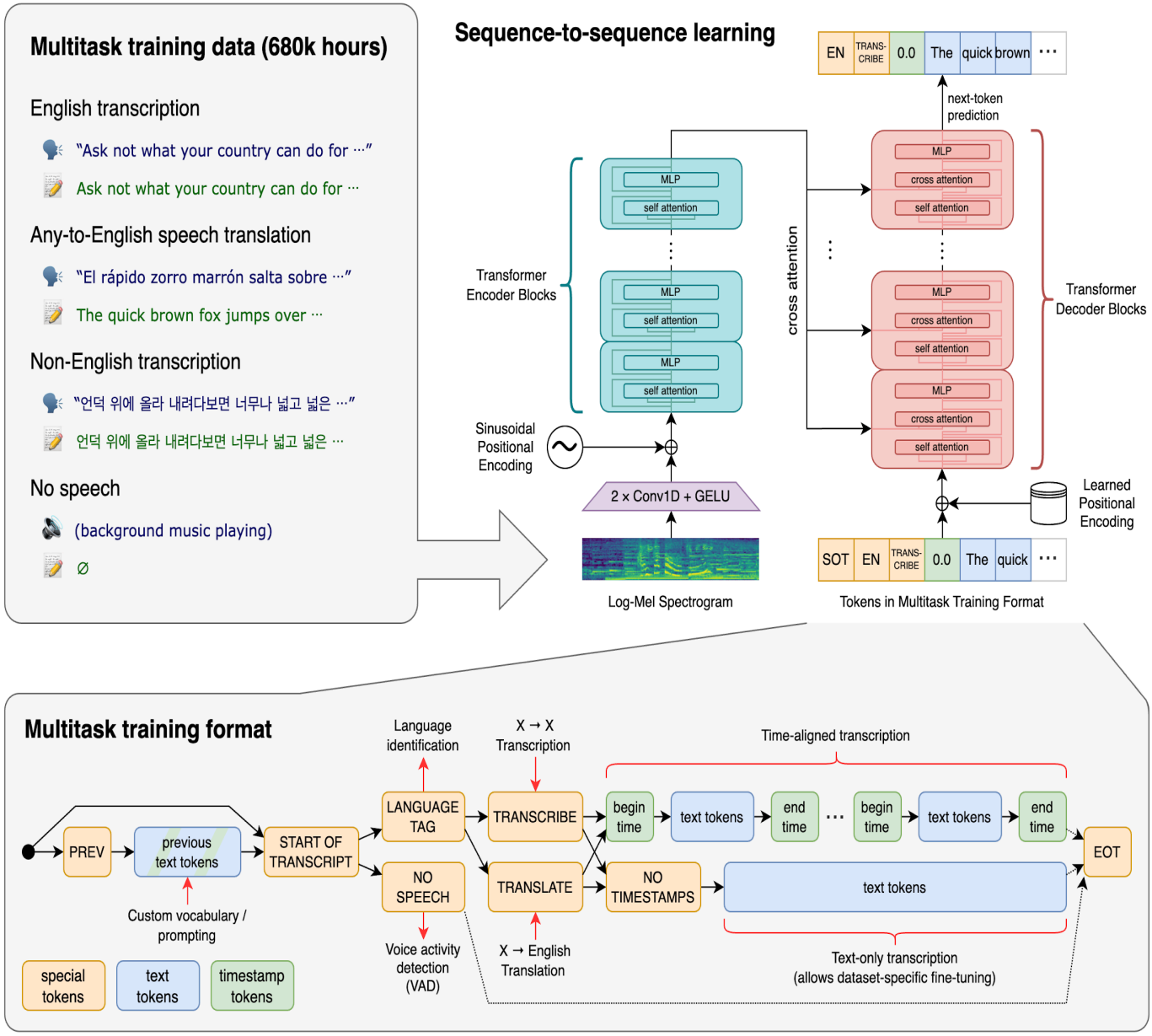
Additional dependencies such as ffmpeg and rust may need to be installed based on the system requirements.

Available Models and Languages

Whisper provides several model sizes optimized for different applications and languages. The table below summarizes the available models along with their specifications:

Size	Parameters	English-only model	Multilingual model	Required VRAM	Relative speed
tiny	39 M	tiny.en	tiny	~1 GB	~32x
base	74 M	base.en	base	~1 GB	~16x
small	244 M	small.en	small	~2 GB	~6x
medium	769 M	medium.en	medium	~5 GB	~2x
large	1550 M	N/A	large	~10 GB	1x

The performance of Whisper varies based on the language and model size. The .en models are optimized for English-only applications and tend to perform better, especially for smaller models.


Multitask training data (680k hours)
English transcription
"Ask not what your country can do for ···"
Ask not what your country can do for ···
Any-to-English speech translation
"El rápido zorro marrón salta sobre ···"
The quick brown fox jumps over ···
Non-English transcription
"언덕 위에 올라 내려다보면 너무나 넓고 넓은 ···"
언덕 위에 올라 내려다보면 너무나 넓고 넓은 ···
No speech
(background music playing)
∅
Sequence-to-sequence learning
Transformer Encoder Blocks
MLP
self attention
Sinusoidal Positional Encoding
2 x Conv1D + GELU
Log-Mel Spectrogram
cross attention
Transformer Decoder Blocks
cross attention
next-token prediction
Learned Positional Encoding
EN
TRANS-CRIBE
0.0
The
quick
brown
SOT
Tokens in Multitask Training Format
Multitask training format
Language identification
X → X Transcription
Time-aligned transcription
PREV
previous text tokens
Custom vocabulary / prompting
START OF TRANSCRIPT
LANGUAGE TAG
NO SPEECH
Voice activity detection (VAD)
TRANSCRIBE
TRANSLATE
X → English Translation
begin time
text tokens
end time
NO TIMESTAMPS
EOT
Text-only transcription
(allows dataset-specific fine-tuning)
special tokens
text tokens
timestamp tokens

Command-line and Python Usage

Whisper can be used both from the command line and within Python scripts for transcription tasks. The command-line usage allows for transcribing speech in audio files, while Python usage provides more flexibility for integration and customization.

1. Command-line Usage

Bash -

whisper audio.flac audio.mp3 audio.wav --model medium

The above command transcribes speech in audio files using the medium model.

2. Python Usage

python
import whisper

model = whisper.load_model("base")
result = model.transcribe("audio.mp3")
print(result["text"])

The Python API enables users to transcribe audio files and provides access to lower-level functionalities for language detection and decoding.

Project Evaluation

Before proceeding with optimization efforts, it's essential to evaluate the current performance of Whisper on feedback calls related to citizen grievances. The evaluation phase involves several key steps:

Data Collection: Gather a diverse dataset of feedback calls in Hindi, English, and Hinglish languages. The dataset should cover various topics, accents, and audio qualities to ensure robust evaluation.

Annotation: Annotate the dataset with ground truth transcripts for each feedback call. The annotations should capture the exact text spoken in the audio recordings.

Evaluation Metrics: Define appropriate evaluation metrics to measure the performance of Whisper. Common metrics include Word Error Rate (WER), Character Error Rate (CER), and Accuracy.

Test Set Preparation: Split the annotated dataset into training, validation, and test sets. The test set should be representative of real-world scenarios and contain a balanced distribution of languages and dialects.

Baseline Performance: Calculate the baseline performance of Whisper on the test set using the defined evaluation metrics. This baseline serves as a reference point for measuring improvements achieved through optimization.

Error Analysis: Conduct a detailed error analysis to identify common transcription errors, language-specific challenges, and areas for improvement. Error analysis helps prioritize optimization strategies and guide model refinement efforts.

Experimentation and Validation

To validate the effectiveness of the optimization strategies, a series of controlled experiments should be conducted:

Experimental Setup: Define clear experimental protocols, including model configurations, training procedures, and evaluation metrics. Ensure reproducibility by documenting all experimental parameters and code implementations.

Hyperparameter Tuning: Perform systematic hyperparameter tuning to optimize model architectures, learning rates, batch sizes, and regularization techniques. Hyperparameter tuning helps identify optimal configurations that maximize transcription performance.
Cross-Validation: Employ cross-validation techniques to assess the generalization performance of the optimized models across different subsets of the dataset. Cross-validation helps mitigate overfitting and provides more robust estimates of model performance.

Statistical Significance Testing: Conduct statistical significance testing to compare the performance of optimized models against baseline approaches. Use appropriate statistical tests, such as t-tests or ANOVA, to determine whether observed improvements are statistically significant.

Qualitative Analysis: Solicit feedback from domain experts and end-users to evaluate the subjective quality of transcriptions generated by optimized models. Qualitative analysis complements quantitative metrics and provides valuable insights into real-world usability and acceptability.

Optimization Strategy

To optimize Whisper for accurately transcribing feedback calls related to citizen grievances, the following strategies can be considered:

Fine-tuning: Fine-tune the existing models on a dataset containing feedback calls in Hindi, English, and Hinglish. Fine-tuning can help the model adapt to domain-specific vocabulary and acoustic characteristics present in citizen grievances.

Language-specific Models: Train language-specific models optimized for Hindi, English, and Hinglish. Language-specific models can capture language-specific nuances and improve transcription accuracy for each language category.

Data Augmentation: Augment the training data with variations such as background noise, speaker accents, and speech rate to improve the robustness of the models.
Model Ensemble: Create an ensemble of multiple models trained on different datasets or with different architectures. Ensemble methods can help mitigate individual model biases and improve overall transcription performance. Augment the training data with synthetic samples generated using techniques such as speed perturbation, noise injection, and pitch shifting. Data augmentation increases the diversity of training examples and improves the model's robustness to variations in audio quality and speaking styles.

Language Model Integration: Incorporate language models trained on domain-specific text data to improve the contextual understanding of feedback calls. Language models can help correct spelling errors, handle out-of-vocabulary words, and enhance transcription accuracy.

Speaker Adaptation: Implement speaker adaptation techniques to personalize the model for individual speakers. Speaker adaptation involves fine-tuning model parameters based on speaker-specific characteristics, such as voice pitch, speaking rate, and accent.

Domain Adaptation: Fine-tune the pre-trained Whisper models on a domain-specific corpus containing feedback calls and related content. Domain adaptation helps the model better capture domain-specific vocabulary, acoustics, and linguistic patterns present in citizen grievances.

Multimodal Fusion: Explore the integration of additional modalities, such as text transcripts or speaker embeddings, to complement the audio-based transcription process. Multimodal fusion techniques leverage complementary information from different modalities to enhance transcription accuracy and mitigate audio-related challenges.

Model Ensemble and Fusion: Develop an ensemble of multiple Whisper models trained with diverse architectures, initialization strategies, and training data. Ensemble methods combine predictions from individual models to produce more accurate and reliable transcriptions. Fusion techniques, such as voting or weighted averaging, can be applied to combine the outputs of ensemble members effectively.

Continuous Improvement and Lifelong Learning

Incorporating mechanisms for continuous improvement and lifelong learning is essential to ensure that Whisper remains relevant, reliable, and effective over time:

Model Retraining and Update Policies: Establish policies and procedures for periodically retraining Whisper models using updated datasets, algorithms, and best practices. Monitor model performance metrics and user feedback to identify opportunities for model refinement and enhancement.

Active Learning and Human-in-the-Loop: Implement active learning techniques and human-in-the-loop systems to iteratively improve Whisper's transcription accuracy and adaptability. Solicit user feedback, annotations, and corrections to refine transcription outputs and address edge cases and ambiguities.

Automatic Model Versioning and Rollback: Automate the versioning and rollback process for Whisper models to ensure seamless deployment and management of model updates and revisions. Maintain a version history and changelog to track model changes and improvements over time.

Benchmarking and Comparative Evaluation: Continuously benchmark Whisper against state-of-the-art speech recognition systems and industry standards to assess its performance relative to competing solutions. Identify areas of strength and weakness and prioritize research and development efforts accordingly.

Experimentation and Innovation: Foster a culture of experimentation and innovation within the Whisper development team by encouraging exploration of novel techniques, algorithms, and methodologies for improving speech recognition accuracy and robustness. Leverage cutting-edge research and emerging technologies to stay ahead of the curve.

User Feedback Mechanisms: Implement robust user feedback mechanisms, such as surveys, ratings, and sentiment analysis, to gather insights into user satisfaction, preferences, and pain points. Use feedback data to inform product roadmap decisions and prioritize feature development and bug fixes.

Cross-Disciplinary Collaboration: Foster collaboration and knowledge-sharing across diverse disciplines, including machine learning, natural language processing, linguistics, and human-computer interaction. Engage with academic researchers, industry experts, and domain specialists to leverage insights from different domains and foster interdisciplinary innovation.

Community Engagement and Participation: Actively engage with the broader community of users, developers, researchers, and stakeholders to solicit input, share knowledge, and co-create solutions. Foster open dialogue, collaboration, and contribution to build a vibrant ecosystem around Whisper and drive continuous improvement and innovation.

Responsible Use and Ethical Governance

Adopting principles of responsible use and ethical governance is paramount to ensure that Whisper's capabilities are harnessed for positive social impact and ethical outcomes:

Ethical Use Policies and Guidelines: Establish clear ethical use policies and guidelines governing the use of Whisper's transcription services to promote responsible and ethical behavior among users and stakeholders. Define acceptable use cases, data privacy requirements, and ethical standards for speech data collection, storage, and processing.

Fairness and Equity Considerations: Integrate fairness and equity considerations into Whisper's decision-making processes, algorithms, and policies to mitigate biases and ensure equitable treatment of users from diverse backgrounds and demographics. Monitor for discriminatory outcomes and take proactive measures to address biases and disparities.

Privacy and Data Protection: Prioritize user privacy and data protection by implementing robust security measures, data encryption techniques, and access controls to safeguard sensitive speech data and user information. Comply with data privacy regulations, such as GDPR, CCPA, and HIPAA, and uphold user rights to data transparency and control.

Algorithmic Accountability and Transparency: Foster algorithmic accountability and transparency by providing users with insights into Whisper's decision-making processes, model architectures, and training methodologies. Enable users to understand and interpret transcription outputs, identify potential biases or errors, and seek recourse in case of algorithmic injustices or discrepancies.

Ethical Review and Oversight: Establish independent oversight mechanisms, such as ethics committees or review boards, to evaluate the ethical implications and societal consequences of Whisper's deployment and usage. Conduct regular audits, impact assessments, and risk analyses to identify and mitigate ethical risks and ensure alignment with ethical principles and values.
Community Engagement and Stakeholder Dialogue: Engage with diverse stakeholders, including civil society organizations, advocacy groups, policymakers, and affected communities, to facilitate dialogue, transparency, and accountability around the ethical use and governance of Whisper. Solicit input, feedback, and recommendations for ethical guidelines and best practices from key stakeholders to inform decision-making and policy development.

Continuous Education and Awareness: Promote awareness and understanding of ethical considerations and responsible AI practices among Whisper users, developers, and stakeholders through training programs, workshops, and educational resources. Empower individuals and organizations to make informed decisions about the ethical implications of using speech recognition technologies and advocate for ethical governance and responsible innovation in the field.

Program

```
import whisper
from pydub import AudioSegment
import os

def convert_to_wav(audio_file):
    # Convert audio file to WAV format
    sound = AudioSegment.from_file(audio_file)
    wav_file = os.path.splitext(audio_file)[0] + ".wav"
    sound.export(wav_file, format="wav")
    return wav_file

def transcribe_audio_to_english(audio_file):
    try:
        model = whisper.load_model("large-v3")
                    result   =   model.transcribe(audio_file,   fp16=False,
task='translate', verbose=True)
        return result["text"]
    except Exception as e:
        print(f"Error transcribing {audio_file}: {str(e)}")
        return None

if __name__ == "__main__":
    audio_file = input("Enter the path of the audio file: ")

    if os.path.exists(audio_file):
        if audio_file.lower().endswith(('.mp3', '.wav', '.ogg')):
            if audio_file.lower().endswith('.mp3'):
                audio_file = convert_to_wav(audio_file)
            transcription = transcribe_audio_to_english(audio_file)
            if transcription:
                print(f"Transcription for {audio_file}: {transcription}")
            else:
                print(f"Failed to transcribe {audio_file}.")
            if audio_file.lower().endswith('.wav'):
                os.remove(audio_file)  # Remove temporary WAV file
        else:
            print(f"Unsupported audio format: {audio_file}")
    else:
        print("File not found. Please provide a valid file path.")
```

OUTPUT

Enter the path of the audio file: /content/_7001847440.mp3
100%|██| 2.88G/2.88G [00:36<00:00, 85.4MiB/s]
Detecting language using up to the first 30 seconds. Use `--language` to specify the language
Detected language: Urdu

[00:00.000 --> 00:04.000] Hello, Hello, Hello, Hello
[00:04.000 --> 00:08.000] Namaskar Sir, I am Prasadni Subhad, from Newmlo, from the Department of Education.
[00:08.000 --> 00:11.000] I am speaking from New Delhi.
[00:11.000 --> 00:15.000] Sir, as we have checked, I am talking to Saurav Kumar.
[00:15.000 --> 00:17.000] Where are you speaking from?
[00:17.000 --> 00:24.000] Sir, I am speaking from New Delhi, Department of Administrative, B.O.M.I.N.D.L. from the Public Revenue Service.
[00:24.000 --> 00:30.000] As we have checked, you have lodged a grievance on 15th June 2013.
[00:30.000 --> 00:31.000] Yes.
[00:31.000 --> 00:36.000] The grievance number is 00029-03. Do you have any information?
[00:36.000 --> 00:38.000] I did it on 15th June.
[00:38.000 --> 00:44.000] You call me in October, call me in November. How will I remember which grievance you are talking about?
[00:44.000 --> 00:50.000] I will check you and tell you what grievances you have lodged. I am talking about this.
[00:50.000 --> 00:52.000] Yes, tell me.
[00:54.000 --> 01:09.000] The grievance was that, sir, in my field, I have repeatedly complained to the Minister of Education that I am constantly paying attention to the use of your upcoming issues.
[01:09.000 --> 01:11.000] Which is in our area.
[01:11.000 --> 01:18.000] In this way, the old government, the 8th district, is troubling.
[01:18.000 --> 01:22.000] Yes, I remember.
[01:22.000 --> 01:23.000] The last cut.
[01:23.000 --> 01:24.000] Yes.
[01:24.000 --> 01:25.000] The grievance was that, sir, you have lodged a grievance on 15th June 2013.
[01:25.000 --> 01:26.000] Yes, I remember.
[01:26.000 --> 01:27.000] Yes.
[01:27.000 --> 01:28.000] Yes.
[01:28.000 --> 01:29.000] I want to know one thing.
[01:29.000 --> 01:34.000] When we report a grievance, why did you make an online portal?

[01:34.000 --> 01:41.000] I will request you on the online portal and then you will often call me to go there and register the complaint again.
[01:41.000 --> 01:43.000] Then what is the point of making this online system?
[01:43.000 --> 01:52.000] But I would like to tell you, sir, the grievance you had lodged, proper action has been taken in its respect and they have received all the information.
[01:52.000 --> 01:53.000] What action?
[01:53.000 --> 01:54.000] They have not done anything.
[01:54.000 --> 02:03.000] It happens every time that I register a complaint and then I get a reply that we have fixed this issue and then it is over.
[02:03.000 --> 02:09.000] But the paper by which the grievance is being filed has been corrected.
[02:09.000 --> 02:11.000] It happens every time.
[02:11.000 --> 02:17.000] Every time I complain, I get a reply every time and the problem persists every time.
[02:17.000 --> 02:19.000] It never gets corrected.
[02:19.000 --> 02:22.000] Have you ever checked whether the problem has been solved or not?
[02:22.000 --> 02:23.000] It never gets corrected.
[02:23.000 --> 02:28.000] It is just that here they have replied to anything and everyone thinks that it must have been solved.
[02:28.000 --> 02:29.000] Yes.
[02:29.000 --> 02:36.000] Sir, the information given by the department that your problem is an issue has been resolved.
[02:36.000 --> 02:37.000] No, it has not been done.
[02:37.000 --> 02:39.000] You reopen it again.
[02:39.000 --> 02:40.000] Okay, sir.
[02:40.000 --> 02:41.000] I will share this in the feedback.
[02:41.000 --> 02:48.000] So, only one answer is being given by the relevant department repeatedly.
[02:48.000 --> 02:50.000] And no conclusion is being drawn.
[02:50.000 --> 02:51.000] It is a big problem.
[02:51.000 --> 02:52.000] Okay, sir.
[02:52.000 --> 02:58.000] And the second thing is that if I am making an online complaint, then it should be a complete online process.
[02:58.000 --> 03:03.000] And I should not have to go somewhere and have to register the complaint again or have to tell.
[03:03.000 --> 03:04.000] Yes, sir.
[03:04.000 --> 03:05.000] I am sharing this in the feedback.
[03:05.000 --> 03:06.000] And I will apologize for the inconvenience.
[03:06.000 --> 03:07.000] So, for that, the rating is poor and average.
[03:07.000 --> 03:08.000] So, what rating should I give?

[03:08.000 --> 03:09.000] Poor.
[03:09.000 --> 03:10.000] Very poor.
[03:10.000 --> 03:11.000] So, you can put that too.
[03:11.000 --> 03:12.000] Yes, sir.
[03:12.000 --> 03:13.000] I am mentioning this in the feedback and I will share it with the department.
[03:13.000 --> 03:14.000] Whatever inconvenience has been caused, I will apologize for that.
[03:14.000 --> 03:15.000] Sir, I have a question.
[03:15.000 --> 03:16.000] Yes, sir.
[03:16.000 --> 03:17.000] I have a question.
[03:17.000 --> 03:18.000] Sir, I have a question.
[03:18.000 --> 03:19.000] Sir, I have a question.
[03:19.000 --> 03:20.000] Yes, sir.
[03:20.000 --> 03:21.000] I would like to apologize for the inconvenience.
[03:21.000 --> 03:22.000] Thank you.
[03:22.000 --> 03:23.000] Thank you for giving me proper feedback.
[03:23.000 --> 03:24.000] It was your day.
[03:24.000 --> 03:25.000] Thank you.
[03:25.000 --> 03:26.000] Thank you.
[03:26.000 --> 03:27.000] Thank you.

Transcription for /content/_7001847440.wav: Hello, Hello, Hello, Hello Namaskar Sir, I am Prasadni Subhad, from Newmlo, from the Department of Education. I am speaking from New Delhi. Sir, as we have checked, I am talking to Saurav Kumar. Where are you speaking from? Sir, I am speaking from New Delhi, Department of Administrative, B.O.M.I.N.D.L. from Public Revenue Service. As we have checked, you have lodged a grievance on 15th June 2013. Yes. The grievance number is 00029-03. Do you have any information? I had done it on 15th June. You call me in October, call me in November. How will I remember which grievance you are talking about? I will check you and tell you what grievances you have lodged. I am talking about this. Yes, tell me. The grievance was that, sir, in my field, I have repeatedly complained to the Minister of Education that I am constantly paying attention to the use of your upcoming issues. Which is in our area. In this way, the old government, the 8th district, is troubling. Yes, I remember. The last cut. Yes. The grievance was that, sir, you have lodged a grievance on 15th June 2013. Yes, I remember. Yes. Yes. I want to know one thing. When we report a grievance, why did you make an online portal? I will request you on the online portal and then you will often call me to go there and register the complaint again. Then what is the point of making this online system? But I would like to tell you, sir, the grievance you had lodged, proper action has been taken in its respect and they have received all the information. What action? They have not done anything. It happens every time that I register a complaint and then I get a reply that we have fixed this issue and then it is over. But the paper by which the grievance is being filed has

been corrected. It happens every time. Every time I complain, I get a reply every time and the problem persists every time. It never gets corrected. Have you ever checked whether its problem has been solved or not? It never gets corrected. It is just that here they have replied to anything and everyone thinks that it must have been solved. Yes. Sir, the information given by the department that your problem is an issue has been resolved. No, it has not been done. You reopen it again. Okay, sir. I will share this in the feedback. So, only one answer is being given by the relevant department repeatedly. And no conclusion is being drawn. It is a big problem. Okay, sir. And the second thing is that if I am doing an online complaint, then it should be a complete online process. And I should not have to go somewhere and have to register the complaint again or have to tell. Yes, sir. I am sharing this in the feedback. And I will apologize for the inconvenience. So, for that, the rating is poor and average. So, what rating should I give? Poor. Very poor. So, you can put that too. Yes, sir. I am mentioning this in the feedback and I will share it with the department. Whatever inconvenience has been caused, I will apologize for that. Sir, I have a question. Yes, sir. I have a question. Sir, I have a question. Sir, I have a question. Yes, sir. I would like to apologize for the inconvenience. Thank you. Thank you for giving me proper feedback. It was your day. Thank you. Thank you. Thank you.

Conclusion

In conclusion, optimizing Whisper for accurately transcribing citizen grievances requires a systematic approach involving data preprocessing, model training, and evaluation. By leveraging the capabilities of Whisper and employing optimization strategies tailored to the task requirements, significant improvements in transcription accuracy can be achieved, thereby enhancing the tool's utility in addressing citizen grievances effectively.

This project report outlines the evaluation and optimization of Whisper, an open-source speech-to-text transcription tool, for accurately converting feedback calls related to citizen grievances into English text. The report provides insights into Whisper's architecture, available models, usage, and optimization strategies aimed at improving transcription accuracy for multilingual speech inputs.