Speech-to-Text with Named Entity Recognition (STT + NER)

(This document primarily covers the project's purpose, challenges, and future plans. You can learn about the project's technical codes, processes, and results through the readme file)

Sections:

About The Project

The project was implemented to create a system for converting speech to text (STT) and identifying named entities (NER) in Uzbek language. The STT model converts audio files to text, then the NER model extracts named objects such as person, place, organization from this text. Special training processes and models adapted to the Uzbek language were created in the project.

Main Components:

1. Speech-to-Text: Obtaining text form of audio data in Uzbek language.
2. Named Entity Recognition: Identifying named objects such as places, people, organizations from text.
3. Pipeline: Combining two models and obtaining entities from audio

First Step:

Before starting the project:

- Previously created projects, their performance quality, and open data were thoroughly studied. (uzbekvoice.ai, aisha.group)
- Numerous articles were studied regarding model and dataset selection
- Overall plan was developed. Resources and datasets for use were reviewed.

  This process took 4-5 days

Model, Problem and Solution:

STT Model:

For creating the speech to text model, **Whisper-base** model was selected for fine-tuning, and **Common Voice 17.0** dataset was chosen

Why whisper-base and Common Voice 17.0?

Model:

- Trained for Uzbek language as well
- Relatively small model and easy to work with
- Many resources and manuals available for working with the model

Dataset:

- Large dataset in Uzbek language
- 17.0 is its latest released collection

Encountered challenges:

The main challenge during the project was the lack of resources (GPU, RAM, DISK). In the first attempt, processes were canceled due to insufficient disk memory

Applied Solution:

The dataset planned for training was split into two parts. And it was decided to train twice. The first obtained model worked with large errors and didn't give us the required result.

In the second stage, the incomplete model was trained again (for the second part of the dataset). The result of the obtained model was satisfactory

As a result, we had two models: .

1. Working with errors, created for training the second model - **whisper-uz** (archived)
2. Retrained based on incomplete model - **whisper-uz-v2**

NER Model:

Adapting the Named Entity Recognition model for Uzbek language was relatively difficult and time-consuming. **Roberta-base** was chosen for fine-tuning, and **uzbek_ner** dataset was selected

Why Roberta-base and uzbek_ner?

Model:

- One of the few models trained for NER in Uzbek language .
- Relatively small model and easy to work with.

Dataset:

- Dataset in JSON format and user-friendly
- No unnecessary columns, doesn't take long to load and process

Encountered challenges:

The main problem was observed in the dataset and model tokenizer, errors in the dataset and deviations in its values negatively affected model quality.

Applied Solution:

Base model's tokenizer was abandoned and Custom Tokenizer was created using dataset containing 130,000 rows: jamshidahmadov/uz_tokenizer

You can see the difference between Base Tokenizer and Custom Tokenizer through notebook

Our obtained model didn't give high accuracy but we created the base model and Tokenizer, conducting pre-train process through finding new and quality datasets in the future will give good results.

Final Result

- **STT Model:** Word Error Rate (wer) ~ 30**.**
- **NER Model:** Precision ~ 97%

Future Plans

Participate in venture fund or startup competitions

For STT model:

- Prepare model for dialects as well
- Train large datasets with large resources

For NER model:

- Find or artificially create quality dataset for model, train it for model:
    - GPT api
    - Translator
    - Data Augmentation

Contacts: Jamshid Ahmadov

Linkedin - linkedin.com/in/jamshid-ds

Telegram - @jamshidds

Gmail - ahmadovv54@gmail.com