

CNG 495
Cloud Computing

Fall - 2023
Term Project Progress Report I

Autospeech

Muhammad Somaan 2528404

Mohamed Eldeeb 2643120

Topic

Android automatic voice translator app using Google Cloud services.

Project Explanation

The aim behind this project is to provide an easy and simple method of communicating with people using different languages. The app will allow capturing voice input, translating it, and playing the translated voice output. It is also planned to have a live speech mode, which would auto-translate in real time without user input (similar to transcription). The uses of this project are many, such as providing a communication method for tourists in a foreign country, or acting as an educational tool for second language learners.

Continuous speech might sometimes heavily affect the already translated words of a sentence. The live speech mode offers a solution for this by allowing automatic translation and transcribing of the voice input, which would result in a more comprehensible speech output. Here is an example of this feature:

Transcribed sentence	Input voice (EN)	Output text (TR)
t1	the quick fox jumps...	hızlı tilki atlar...
t2	the quick fox jumps over...	hızlı tilki üzerinden atlıyor...
t3	the quick fox jumps over the lazy dog.	hızlı tilki tembel köpeğin üzerinden atlar.

As can be seen, the translated speech differs greatly after each word is added. Because of this, when the output voice is spoken, a text transcription of the translated sentence will be displayed, and if there was any text that needed retranslation (e.g. t1 to t2, or t2 to t3), it will be re-spoken separately of the already spoken words. After the input sentence is finished (or is structurally complete), the fully translated voice output can be re-spoken.

This feature tries to mimic the behaviour of human translators, who would often wait until the sentence they are hearing is finished, or they would repeat part of it to re-explain it better.

Cloud Delivery Models

The following Google Cloud SaaS (Software as a Service) components will be used in this project:

- Speech-to-Text API: Converts audio to text by applying powerful neural network models.
- Translation API: Integrates text translation into a website or application.
- Text-to-Speech API: Synthesizes natural-sounding speech by applying powerful neural network models.

Diagrams

The following diagram (figure 1) describes the data flow between the application and cloud services.

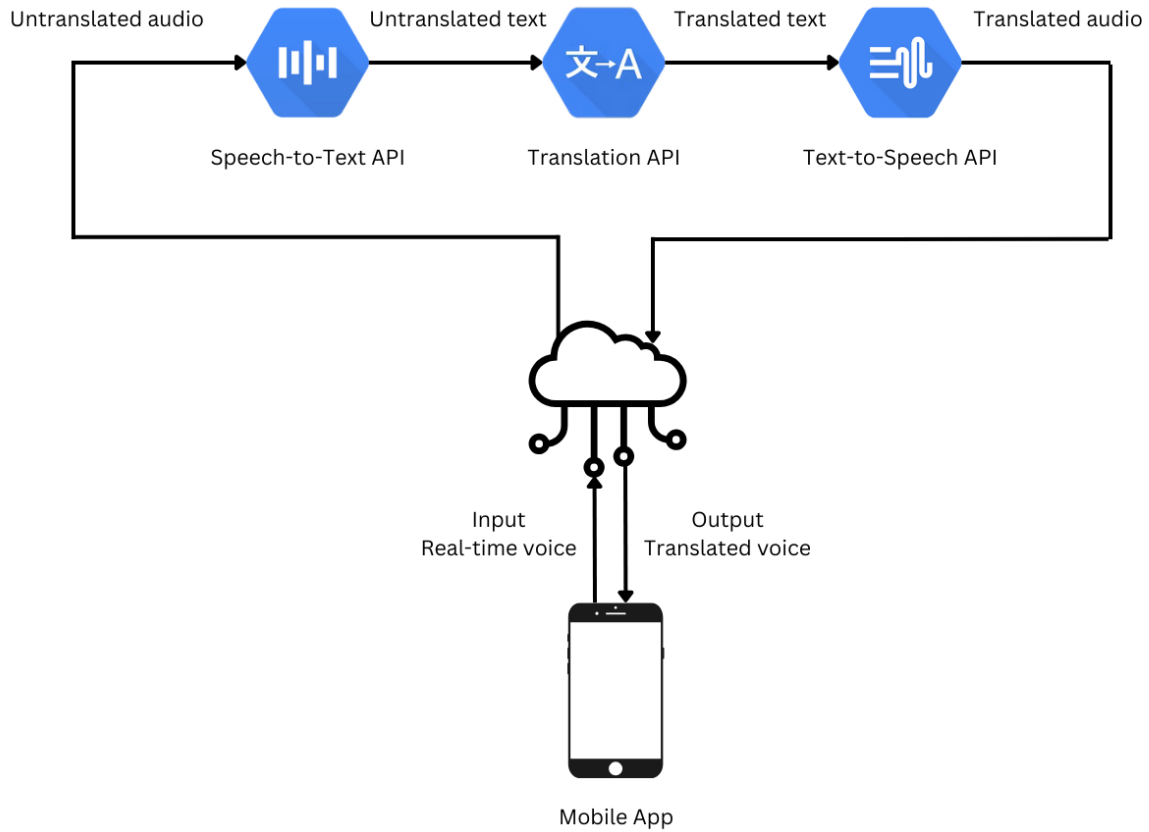


Figure 1: Data flow between application and cloud services

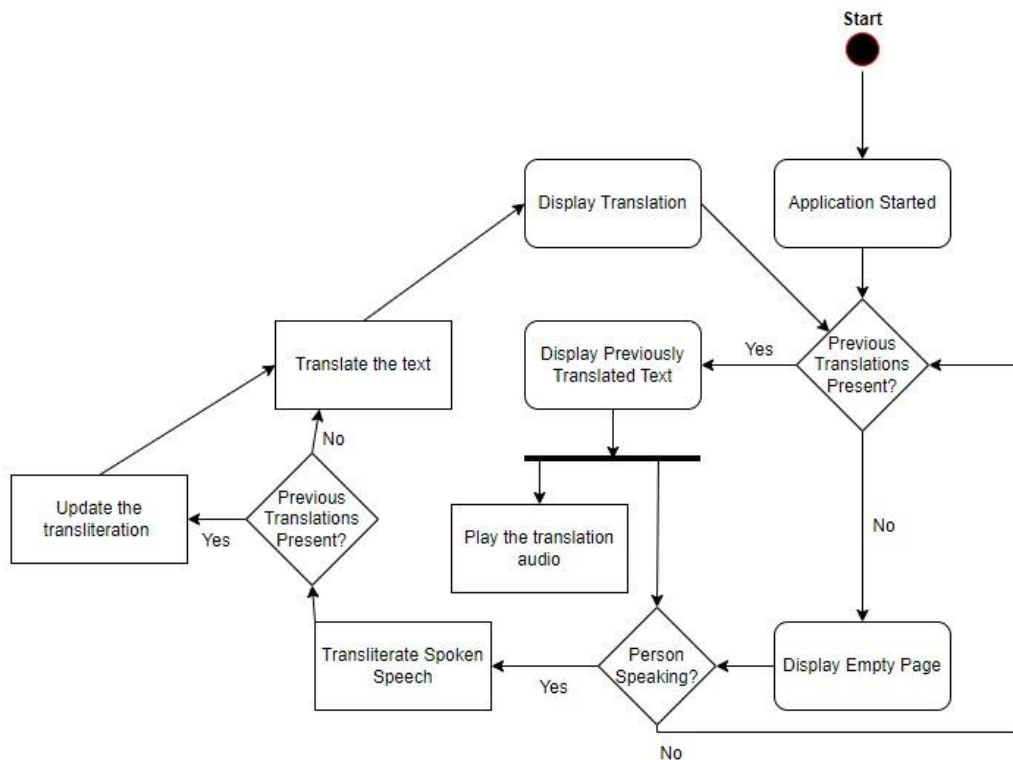


Figure 2: Process of transliterating and translating speech input

Figure 2 is a computation diagram that shows the process of transliterating and translating speech input.

Expected Contribution of Team Members

Here is the expected contribution of each team member:

Somaan:

Creating the UI of the application as well as interactive features. Implementing the speech-to-text and translation parts of the app.

Eldeeb:

Communicating with the cloud services and formatting data. Implementing the text-to-speech part of the app, in addition to the live speech mode.

Milestones Achieved

27 November 2023 - 03 December 2023:

Somaan & Eldeeb:

- Watched tutorials on setting up a working app in Android Studio.
- Created an empty project on Android Studio.
- Researched Google's built-in APIs in Android Studio for Speech-to-Text, Translation, and Text-to-Speech.

04 December 2023 - 10 December 2023:

Somaan:

- Started testing Google's speech-to-text service in Android Studio.
- Searched for example usage for Speech-to-Text service.[4]
- Created a UI to manage speech recognition.

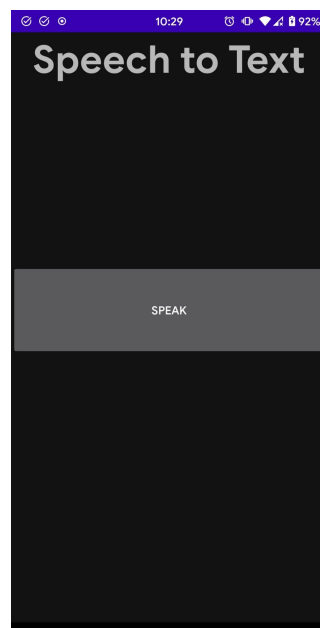


Figure 3: Speech Recognition default UI

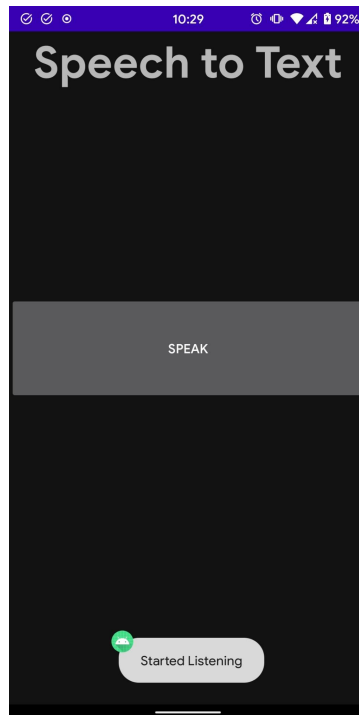


Figure 4: Speech Recognition UI after clicking Speak

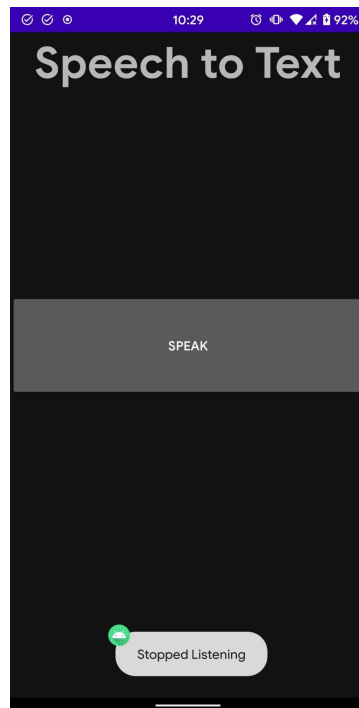


Figure 5: Speech Recognition UI after clicking Speak again

- Figure 3 shows the default test UI for speech recognition.
- Figure 4 shows the message popup on the screen when the user presses the speak button.
- Figure 5 shows the message popup on the screen when the user presses the speak button again.
- The Speak button, when pressed, starts listening to what the user is speaking and keeps on listening until the user presses the Speak button again, upon which the recording is stopped.
- I had some issues in making it properly recognize the speech and then output a text, but the general skeleton for it is done, and I will look into it in the future.

Eldeeb:

- Searched for usage examples of Google Cloud APIs in Android with Kotlin [5].
- Initial tests for Google Cloud Translate API.

11 December 2023 - 17 December 2023:

Somaan:

- Started testing Google's translate service in Android Studio.
- Searched for example usage for Google's translate API.[5]
- Searched for example usage for detecting the language of the text.
- Created a UI to manage translation.

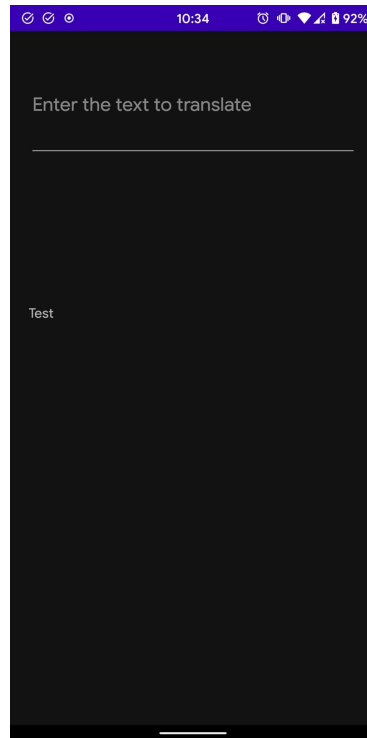


Figure 6: Default Translate UI

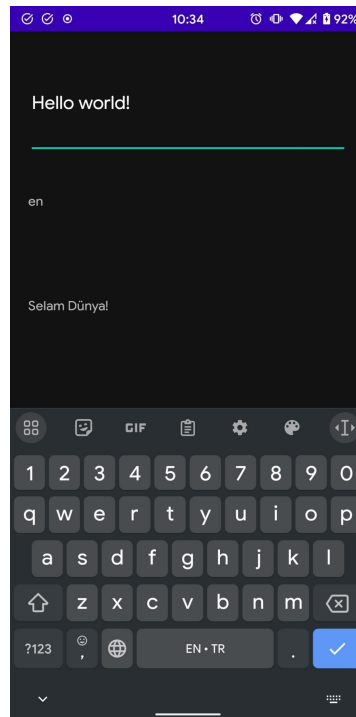


Figure 7: Working Example of Translate UI

- Figure 6 shows the default test UI for translation.
- Figure 7 shows working example of translation, where the top text is the text which is going to be translated, the middle text is the language code for text input (automatically detected), and the bottom text is the translation in Turkish.
- The translation feature works as we want it to and is accomplished by using the Google Cloud API, so it does not rely on any model to be downloaded or saved to the device. The only downside to this is that the app requires an active internet connection to utilize this feature.

Eldeeb:

- Setup the project for deployment of local credentials file (for Cloud API).
- Tested Google Cloud Translate API in real time.
- Various UI fixes (navigation, margins, etc.)

Milestones Remained

18 December 2023 - 24 December 2023:

Somaan:

- Work on fixing the issues in speech-to-text and further research on how to improve its efficiency and accuracy.

Eldeeb:

- Implement integration for Google Cloud text-to-speech.

25 December 2023 - 31 December 2023:

Somaan:

- Remove any test code or local APIs that are provided by Google in Android Studio and shift to Google Cloud's APIs.
- Start making the UI better.
- Implement and test the features using solely APIs provided by Cloud, removing any local dependency.

Eldeeb:

- Implement live speech mode and its UI.

1 January 2024 - 7 January 2024:

Somaan:

- Unify the app into 1 main UI, which will use all the services in our project proposal.
- Properly test the app for any bugs or issues with UI.
- Test the app to learn the limitations of the system.

Eldeeb:

- Write documentation and a guide.
- Debug live speech mode and perform intensive testing.

Deliverables after project is completed:

- Code for Android app.
- Brief usage guide.

GitHub repository link:

<https://github.com/SutandoTsukai181/AutoSpeech>

References

1. Google. (n.d.). Cloud Speech-to-Text API.
<https://console.cloud.google.com/marketplace/product/google/speech.googleapis.com>
2. Google. (n.d.). Cloud Translation API.
<https://console.cloud.google.com/marketplace/product/google/translate.googleapis.com>
3. Google. (n.d.). Cloud Text-to-Speech API.
<https://console.cloud.google.com/marketplace/product/google/texttospeech.googleapis.com>
4. Google. (n.d.). Speech Recognizer
<https://developer.android.com/reference/android/speech/SpeechRecognizer>
5. Cansu Yeksan Aktaş. (2019). How to use Google Translate API in Android Studio projects?
<https://medium.com/@yeksancansu/how-to-use-google-translate-api-in-android-studio-projects-7f09cae320c7>