Industrial Attachment End-Term Report

For reporting period: 1st Feb –7th May 2021

Name: Geet Jethwani

Matriculation Number: A0215395B

Course: Intelligent Systems

Supervisor: Ritesh Munjal

Company: Schlumberger

Table of Content

Acknowledgement

The internship I had with Schlumberger was a great opportunity for learning and professional development. I would like to express my gratitude to everyone at Schlumberger SWTC (Singapore Wells Testing Centre) for their guidance and support over the course of my internship.

I would like to especially thank my supervisor, Mr. Ritesh Munjal for his unwavering support and guidance throughout my internship stint. I was given many invaluable opportunities to learn and develop my technical and non-technical skills. I am also grateful to all the people at SWTC who guided me through the various processes to ensure that I was able to complete my tasks successfully.

This internship was a valuable experience and the skills that I have learnt will be beneficial to my career in the future.

Introduction

This final report covers the duration from 1st Feb –7th May 2021 of my Industrial Attachment at Schlumberger Ltd. Schlumberger Limited is an oilfield services company. Schlumberger employees represent more than 140 nationalities working in more than 120 countries. Schlumberger has four principal executive offices located in Paris, Houston, London, and The Hague.[1]

In this report, I will present an overview of the industry and company background, description of completed project, reflections on the challenges encountered during the project and solutions implemented to overcome them.

Company Background

History:

Schlumberger was founded in 1926 by two brothers Conrad and Marcel Schlumberger from the Alsace region in France as the Electric Prospecting Company (French: Société de prospection électrique).

The company recorded the first-ever electrical resistivity well log in Merkwiller-Pechelbronn, France in 1927.[1]

Today, Schlumberger supplies the petroleum industry with services such as seismic acquisition and processing, formation evaluation, well testing and directional drilling, well cementing and stimulation, artificial lift, well completions, flow assurance and consulting, and software and information management. The company is also involved in the groundwater extraction[1] and carbon capture and storage industries

About:

Schlumberger is a technology company that partners with customers to access energy. Schlumberger employees, representing over 160 nationalities, are providing leading digital solutions and deploying innovative technologies to enable performance and sustainability for the global energy industry. With expertise in more than 120 countries, we collaborate to create technology that unlocks access to energy for the benefit of all.[2]

SWTC Flow loop – Schlumberger Singapore

The SWTC multiphase flow loop is operational since April 2016. This state-of-the-art flow loop has been designed to serve both the manufacturing (FAT, CWT) and Engineering (NPD) activities of SWTC.

Its main purpose is to evaluate the performance of MPFMs by circulating precise mixtures of oil, water and gas at various flow rates and pressures.

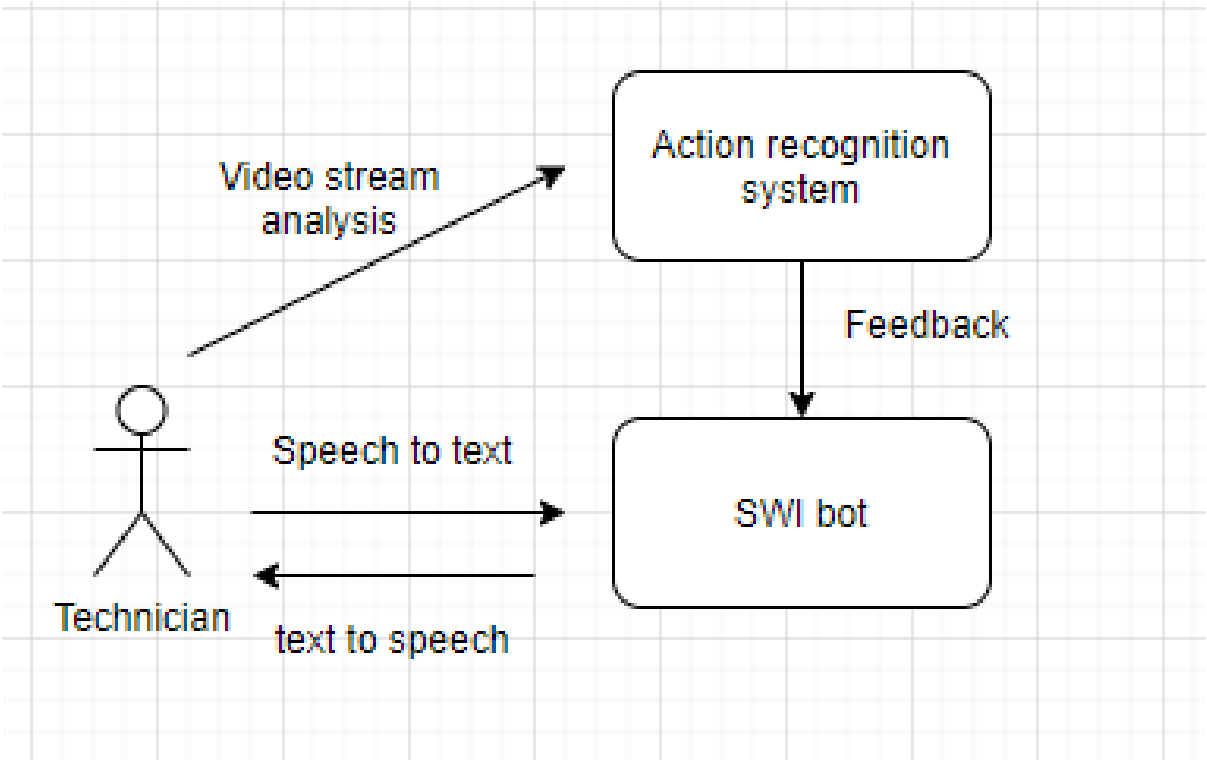
Description of project components:

Problem Overview:

When new technicians join Schlumberger, they are asked to perform certain tasks or have a training that will enable them to perform tasks relevant to their course of action or job function. However, going through the documents manually and understanding the steps can be cumbersome. Moreover, there is no evaluation criteria of how well the action is being performed.

Thus, the idea is to build an intelligent training mechanism, which includes a chatbot that a technician can talk to and an action recognition system through live video stream.

This would be not just for the new trainees but also for existing employees. As the SWI bot would reinforce all the steps and procedures required to be completed for a particular task and an action recognition system would be able to monitor whether they are performing the actions correctly.

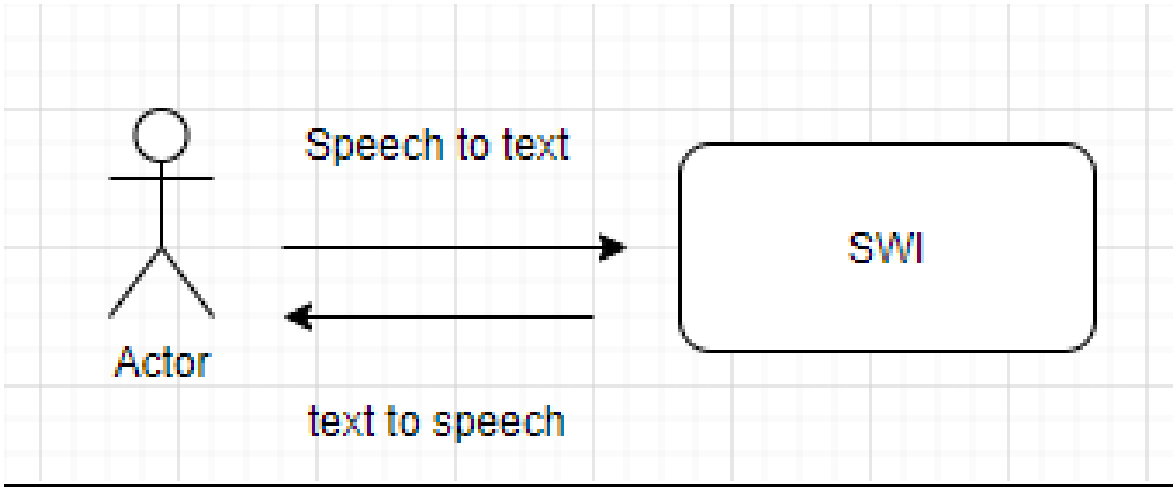


Project Objectives:

The task assigned to me was to cover the SWI bot part, where the technician could interact with the chatbot to fetch the documents required for training and then have out read out to him.

This feature basically includes components where in a chat interface had to be built for the technicians to interact with. The Chat UI had to include 2 ways of interacting with the user:

i) Text: User should be able to type the name or relevant keyword to fetch the document
ii) Speech: should be able to interact with the bot trough speech



The following project was completed over the past 3 months (01st Feb-07th May 2021):

1. **Chatbot**
   a. Overview

      The Chat UI had to include 2 ways of interacting with the user:

      i) Text: User should be able to type the name or relevant keyword to fetch the document
      ii) Speech: should be able to interact with the bot trough speech

   b. User Interface :

      The user interface was built using :

      i. Web Dev : HTML , CSS , JavaScript
      ii. Backend Script : Python
      iii. Web app framework : flask

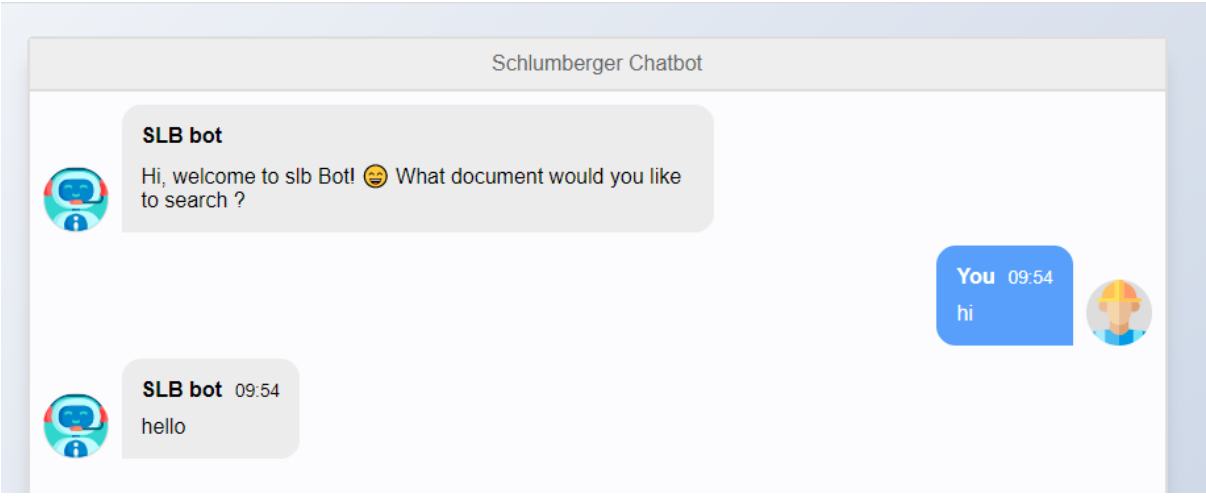


Figure 1: Chat UI.

2. Speech Recognition and synthesis

a. Overview

The user first speaks into the microphone. This is achieved with the help of Google Web Speech API.

The Speech-to-Text API enables developers to convert audio to text in over 120 languages and variants, by applying powerful neural network models in an easy-to-use API. This is a very powerful speech to text recognition API toolkit. And then this recognized text is sent to the Chat UI (the entire flow has been explained above).

Speech recognition involves receiving speech through a device's microphone, which is then checked by a speech recognition service against a list of grammar.
When a word or phrase is successfully recognized, it is returned as a result
Approaches considered :

- Using the python SpeechRecognition library (Google Web Speech API)
- Web Speech API

After reception of the bot response the synthesis of the speech takes place in order to handle to text to speech. This is done with the help of Pyttx3 library.
Pyttsx3 is a text-to-speech conversion library in Python.

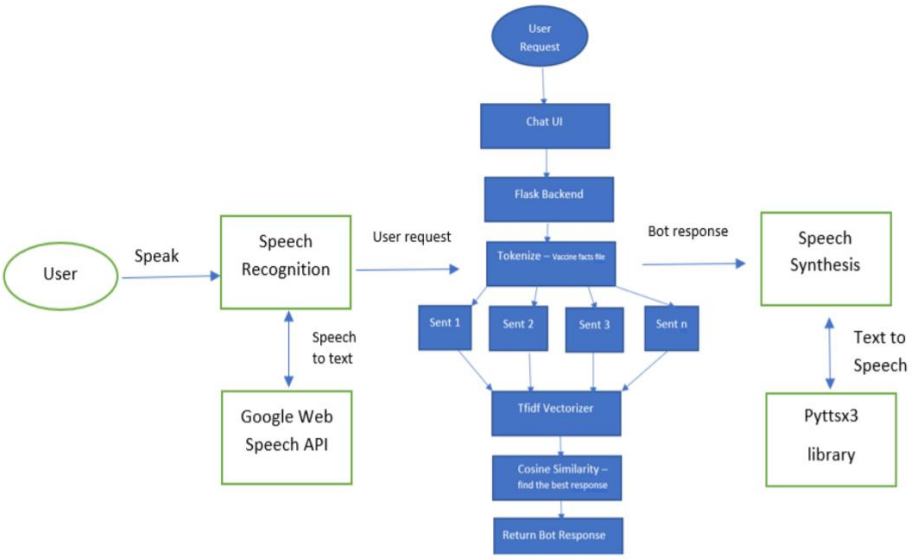
Unlike alternative libraries, it works
offline and is compatible with both Python 2 and 3. An application invokes the pyttsx3.init() factory function to get a reference to a pyttsx3. Engine instance. it is a very easy to use tool which converts the entered text into speech.

The pyttsx3 module supports two voices first is female and the second is male which is provided by “sapi5” for windows.

However , in the final implementation web speech API is used.

It supports three TTS engines:
• sapi5 – SAPI5 on Windows
• nsss – NSSpeechSynthesizer on Mac OS X
• espeak – eSpeak on every other platform

The processed speech is then sent as an output where the bot repeats the words in the form of speech from the microphone.



3. Document Search

a. Text Search

Firstly, the user sends a request to the chatbot UI. It could be anything ranging from a question
to just a thought.
The request is then sent to the chat bot UI which processes the request text information and extracts it which would then be sent to the backend.
The request is then sent to the flask backend where the request information from the frontend
is processed. Flask is a web server application which is used to host the backend of a website
where URLs are exposed so that the frontend can connect with it.
The sentence or request sent by the user is then tokenized or transformed into a list of words with the help of NLTK library of python. This helps in word embedding generation and breaks
down the sentence into works which can then be lemmatized and stemmed to better understand the structure of the sentence.

Different words of the sentences are then sent to the TF-IDF vectorizer to obtain the most relatable sentence from the data source based on the tokenization of the sentences. TF-IDF stands for “Term Frequency — Inverse Document Frequency”. This is a technique to quantify a word in documents, we generally compute a weight to each word which signifies the importance of the word in the document and corpus. This method is a widely used technique in Information Retrieval and Text Mining. When we are vectorizing the documents, we check for each word count. In worst case if the term doesn’t exist in the document, then that TF value will be 0 and in other extreme case, if all the words in the document are same, then it will be 1. The final value of the normalized TF value will be in the range of [0 to 1]. 0, 1 inclusive.

Cosine similarity is a measure of similarity between two non-zero vectors of an inner product space.

It is defined to equal the cosine of the angle between them, which is also the same as the inner product of the same vectors normalized to both have length 1. The cosine of 0° is 1, and it is less than 1 for any angle in the interval (0, π] radians. It is thus a judgment of orientation and not magnitude: two vectors with the same orientation have a cosine similarity of 1, two vectors oriented at 90° relative to each other have a similarity of 0, and two vectors diametrically opposed have a similarity of -1, independent of their magnitude. The cosine similarity is particularly used in positive space, where the outcome is neatly bounded in [0,1].

The name derives from the term "direction cosine": in this case, unit vectors are maximally "similar" if they're parallel and maximally "dissimilar" if they're orthogonal (perpendicular).

b. formulae

Term frequency formula:

term frequency	$f_{t,d} \Big/ \sum_{t' \in d} f_{t',d}$

Inverse document frequency formula:

inverse document frequency	$\log \frac{N}{n_t} = -\log \frac{n_t}{N}$

Figure 6: TF-IDF formulae.

c. Procedure

i. Filter user text using stop words and stemming
ii. Split the user text into different words
iii. Search for the filtered user text on the page
iv. If user text is found add highlight annotation (PyMuPDF ,fitz)
v. If all instances are found in the filtered text save the document with the highlighted references.

The added advantage of splitting the filtered user text and finding all utterances is as follows :

The searched text need not exactly match the procedure name.

d. Stop Words

Using Stop words :
A list of stop words had to be defined to filter out the user text .
Stop words used :
Stop words downloaded from nltk library

- Examples : [‘I', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you’re”]

Custom stop words added based on the application requirement

- Examples : ['open', 'save', 'document', 'please', 'show', 'could', 'about', 'search']

e. Stemming and Lemmatization

Lemmatization is the process of converting a word to its base form.

The difference between stemming and lemmatization is, lemmatization considers the context and converts the word to its meaningful base form
stemming just removes the last few characters, often leading to incorrect meanings and spelling errors.
Thus, lemmatization is used to convert the words to its base form.
E.g. windows -> window , procedures -> procedure

4. PDF Audio Operations

a. Overview

This module is about PDF reader being utilized as a Speech to text framework that would read out the text of the PDF. The following steps have been used to perform this operation :

i) Conversion of PDF file to text file
ii) Data cleaning
iii) Addition of SSML tags
iv) Conversion of text to audio

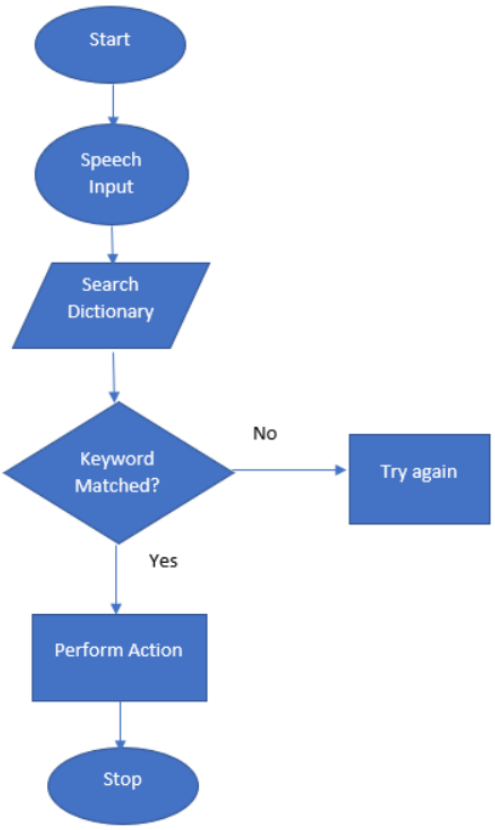
The entire process was carries out using a python script and balabolka application. As balabolka makes it easier to interpret PDFs and convert to audio , this was mainly used for the conversion.

5. Application Control Through Speech

The application can be controlled through speech, in the sense that, if the user wants to navigate through the pdf or play / stop the audio , he can perform those operations through speech. The following keywords have been used. Every keyword has a different action associated with it:

1) Next page : slides down the pdf to the next page
2) Play : plays the procedures
3) Stop / Pause : pauses the procedure
4) Back/previous page : moves back to the previous page
5) Next(step) : scrolls down to the next step
6) Back : moves slightly up , to the previous
7) Start clock : starts the clock

These are few of the voice operations performed by the user .



6. **Deployment**:

Docker : OS-level virtualization to deliver software in packages called containers. Containerized the flask app to deploy it on any platform required.

Initially Deployed it on Google Cloud Platform. Currently Deployed on Azure (web app).

Steps to Deployment :

- Build a docker image
- Push it to docker hub
- Pull the image into azure
- Run the azure web service

Lessons Learnt & Possible Future Application

Working on this project have allowed me to pick up some important lessons. These lessons, in my opinion, are the most important take-away during my time in the company.

While developing the chatbot and designing the UI , I have realized that it is important to communicate efficiently. Usually these projects involve multiple parties and it is essential for us to be able to communicate our ideas and opinions effectively to the other parties. These could be how the layout of the website should look like or how to optimize the code to make the retrieve the data faster. Additionally, I have also learnt to keep an open mind and be adaptive. More often than not, there will be changes made to the projects and we will have to implement these changes in a given amount of time. In order for us to implement these changes, we will need to adapt quickly and learn new processes if required. From this, I have learnt that communication and teamwork is key to delivering a project on time.

Moreover, interning in an industry outside my major has taught me that the eagerness and willingness to learn is an important value. When I first started my internship, I did not have know much about manufacturing and how I would be able to build a project, that would have an positive impact. However, after reading more documentation online and asking my mentors for guidance, I was slowly able to carry out the tasks assigned to me. Without the willingness to learn, I would not have been able to contribute to the projects as much as I have today. Having this mindset will be useful as I embark on various career paths in the future.

Through these experiences, I have learnt to become more confident and independent in the way I work while being receptive to advice. I have also learnt to be flexible and think out of the box to solve problems strategically. These experiences and lessons that I have picked up during this internship will definitely be valuable for my future career.

References

[1] https://en.wikipedia.org/wiki/Schlumberger

[2] https://www.linkedin.com/company/schlumberger

[3] Edward C. Baig, USA TODAY (17 April 2015). "More Big Oil layoffs: Schlumberger axes 11K jobs". USA TODAY. Retrieved 28 August 2015.

Weekly Journal

	Key Learning Point
Week 1 1 – 5 Feb 2021	• Problem Understanding • Developing a basic chat interaction program .
Week 2 8 – 12 Feb 2021	• Problem Understanding • Developing a basic chat interaction program • Understanding algorithms such as tf-idf • Explored options to integrate for front end • Explored available chatbot builders in the market.
Week 3 15 – 19 Feb 2021	• Built front end using HTML , CSS , JS • Explored ways for speech synthesis • Implemented basic speech recognition using the systems microphone
Week 4 22 – 26 Feb 2021	• Completed chatbot UI/frontend • Implemented speech synthesis • SIPP and NEST Training
Week 5 1 – 5 Mar 2021	• Pushed code to Docker • Deployed basic application (without speech) to Google cloud platform • Application status : Deployed
Week 6 8 – 12 Mar 2021	• Improved Speech recognition through Webkit API • Incorporated Speech recognition to bot • Deployed application on azure -
Week 7 15 – 19 Mar 2021	• Improved Speech recognition through Webkit API • Incorporated Speech recognition to bot • Deployed application on azure

Week 8 22 – 26 Mar 2021	• Improved Document searching algorithm by adding stop words, Lemmatization • Build User Interface for rendering PDF and Audio tag together . • Text to speech functionality for Final Mech Complete ! -
Week 9 29 Mar – 2 Apr 2021	• If Document found , opened automatically • Improved speech recognition through dictionary parsing
Week 10 5 – 9 Apr 2021	• Application control through speech. • Worked on building keyword dictionaries to perform different operations
Week 11 12 – 16 Apr 2021	• Now user can start/stop and pause audio through voice commands • Attempts made to reduce user intervention.
Week 12 19 – 23 Apr 2021	• User Application Testing
Week 13 26-30 Apr 2021	• Clocking and auto scrolling • Now user can start/stop and pause clock through voice commands • Also with auto scrolling , user intervention has reduced. • Presentation