Advanced Technology Attachment Programme (ATAP)

Final Project Report

at

Government Technology Agency

Reporting Period

6 Jan 2020 to 30 June 2020

by

Ng Wei Jie, Brandon

Department of Computer Engineering

School of Engineering

National University of Singapore

2020/2021

Project Title: Data Analytics/ Natural Language Processing Engineer
Project ID: AY19142758
Project Supervisor: Sim Mong Cheng, Terence

Summary

AskJamie is Singapore's virtual assistant to provide information to end users directly through chatbots. Currently the Natural Language Processing (NLP) engine in AskJamie is supported by a third party vendor FlexAnswer. With latest advancements in NLP and chatbot technologies, my team at Goverment Technology Agency (GovTech) is looking into better NLP engine alternatives. As AskJamie's NLP engine is migrated, one of the sub-objectives is to provide a platform for users to clean the chatbot's training dataset. The idea came about after observing end users adding poor training data into the chatbot dataset that resulted in poor performances in the chatbot. Therefore, the platform, I am tasked with, aims to provide end users an overview of the chatbot dataset and identify areas of improvement in the knowledge base. The platform will also help users to migrate data from one vendor to another and support regression testing to ensure the chatbot produces the correct response after updates are made to the training dataset.

Subject Descriptors:

I.2.7 Natural Language Processing

I.5.3 Clustering

D.2.11 Software Architectures

Keywords:

Natural Language Processing, Chatbot, Big Data

Implementation Software and Hardware

Scikit-Learn, Gensim, Mallet, Tensorflow, PyTorch, Bert, Albert, Nvidia, NLP, ReactJS, Flask, MongoDB, PySpark, Docker, Kubernetes

Contents

1 Introduction

1.1 Background and Organizational Structure of Host Organization

I am doing my ATAP at Government Technology Agency (GovTech), one of Singapore's statutory board. It is under the Prime Minister's Office (PMO) as shown in Figure 1. It was restructured from the former entities Infocomm Development Authority of Singapore (IDA) and Media Development Authority (MDA) in 2016, and officially legislated in Parliament on 18 August 2016.[1]

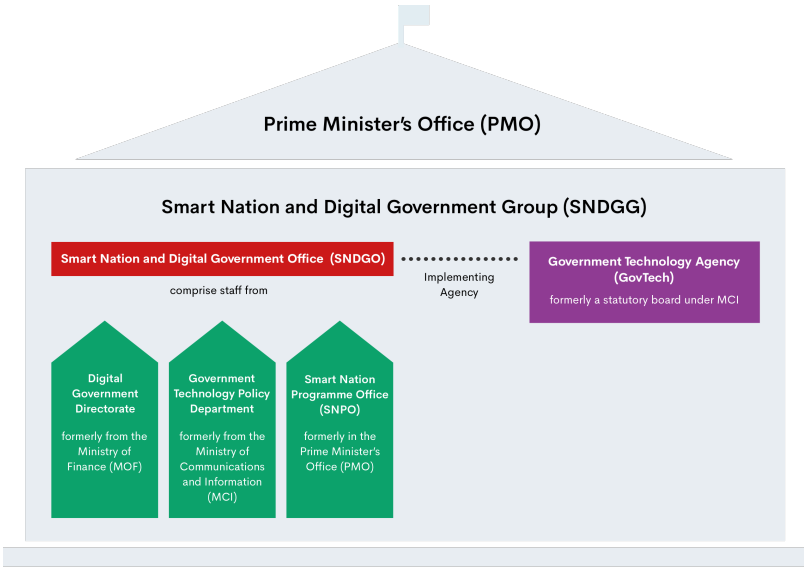

Figure 1: Organisational Chat For Smart Nation and Digital Governmnt Group (SNDGG) In Prime Minister Office[2]

GovTech is created to spearhead Singapore's Smart Nation Initiatives and build capabilities in new technologies to shape the way business is done in the government. It focused on areas of Application Development, Cybersecurity, Data Science, Government ICT Infrastructure, and Sensors & IoT.[2]

1.2 Principal Activities of Host Organization

GovTech outlined Singapore's Digital Government Blueprint (DGB) which is a statement of the Government's ambition to better leverage data and harness new technologies, and to drive broader

efforts to build a digital economy and digital society as part of Singapore's Smart Nation Initiative. Its vision is to create a Government that is "Digital to the Core, and Serves with Heart". This means building user-centric services that cater to citizen's and businesses' needs. Using digital government services will also be easy and secure. Public officers will have opportunities to continually up-skill and adapt to new challenges.[3]

DGB will benefit the public by providing services that have digital signature options, are intuitive, are secured, and are catered for end users' needs. Examples of digital services that are already available and benefiting the public are Moments of Life, Business Grants Portal, and ParkingSG applications.[3]

1.3 Training Programme within Host Organization

GovTech has a leadership-trainee programme TAP which is designed to sharpen and develop technical knowledge and professional skills for fresh graduates. Participants will gain two years of specialist training and be groomed to take on specialist and technology leadership roles within GovTech that can also accelerate their career development.[4]

GovTech has developed a mobile application Learn GovSG where Public Officers can upgrade their skills through online videos. Teams in Govtech also organized workshops occasionally where others can signup and upgrade skills.[5]

1.4 Position of Host Unit within Host Organization

I am in the division Moments of Life (MOL). It is one of the Strategic National Projects under Singapore's Smart Nation Initiative that places citizens at the heart of digital government services at key life moments. It is a suite of services, which supports citizens' needs at key junctures by integrating and bundling services across Government agencies. It currently supports families with children aged 6 and below, and seniors aged 60 and above, by bundling useful services and information on a single digital platform.[6]

2 Training Schedule And Assignments

2.1 Training Schedule By Month For The Entire Training Period

My training schedule is depicted in Table 1. My training schedule is not fixed as there are random task that popped up during my training period.

Table 1: Training Schedule

Task	Metric	Month
Create NLP pipeline to identify conflicts in chatbot dataset	RAM usage and computation time	Jan
Create mockups of dashboard and conduct user testing	Intuitive designs	Feb
Develop dashboard and integrate NLP pipeline in the backend code	Robustness and latency	Mar
Roll out first version of dashboard on AWS	Robustness and latency	Apr
Setting migration tools and testing on the dashboard	Robustness and accuracy	May
Implement pyspark to handle large training datasets and syncing content between dialogflow and dashboard	Scalability	Jun

2.2 Training Assignments Completed in 1st Month

Due to the recent outbreak of Coronavirus in Singapore, my team was tasked to built a chatbot to disseminate information on the Coronavirus situation in Singapore as shown in Figure 2. It was done in 2 days. I was involved in training the chatbot on Dialogflow.

After which, I helped to redesigned the chatbot to give it a more ”Telegram” feel. I also scraped information from MOH website to provide end users latest updates on Coronavirus situation in Singapore. Besides, to ensure that adding new training content does not reduce the chatbot's performance, I helped automated regression testing. Test cases are utterances that my team prepare on Google Sheet and the expected response are compared against the actual response obtained from DialogFlow APIs. The automation tool will pull these test cases from Google Sheet, tests the response with Dialogflow APIs, and populated the results back on Google Sheet.

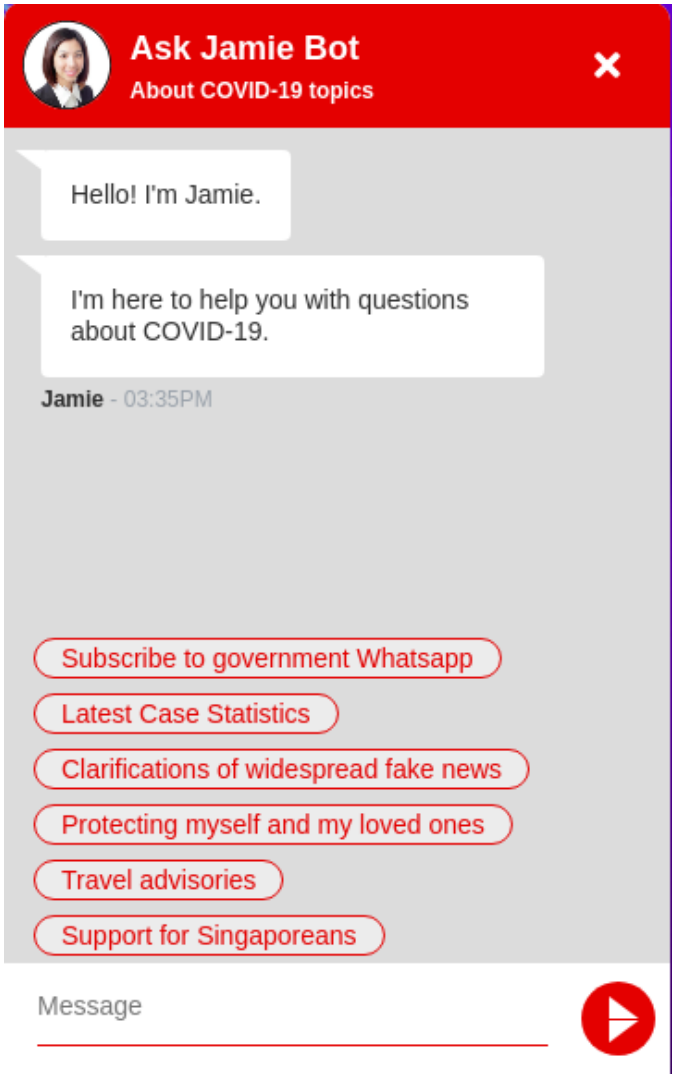

Figure 2: GovSG Chatbot To Disseminate Information On nCov

2.3 Training Assignments Completed in 2nd Month

One of my main task is to create a platform to provide end users an overview of the chatbot dataset and identify areas of improvement in the knowledge base. This platform is a web-based dashboard where user only uploads their dataset. After the dataset is processed, the user can go to the dashboard to view and understand the different issues in the chatbot dataset. Before commencing the software development works, I started by doing up mockups using AdobeXD for the dashboard and did user testing to ensure the dashboard is intuitive enough for end user. The mockup for the dashboard page is shown in Figure 3.

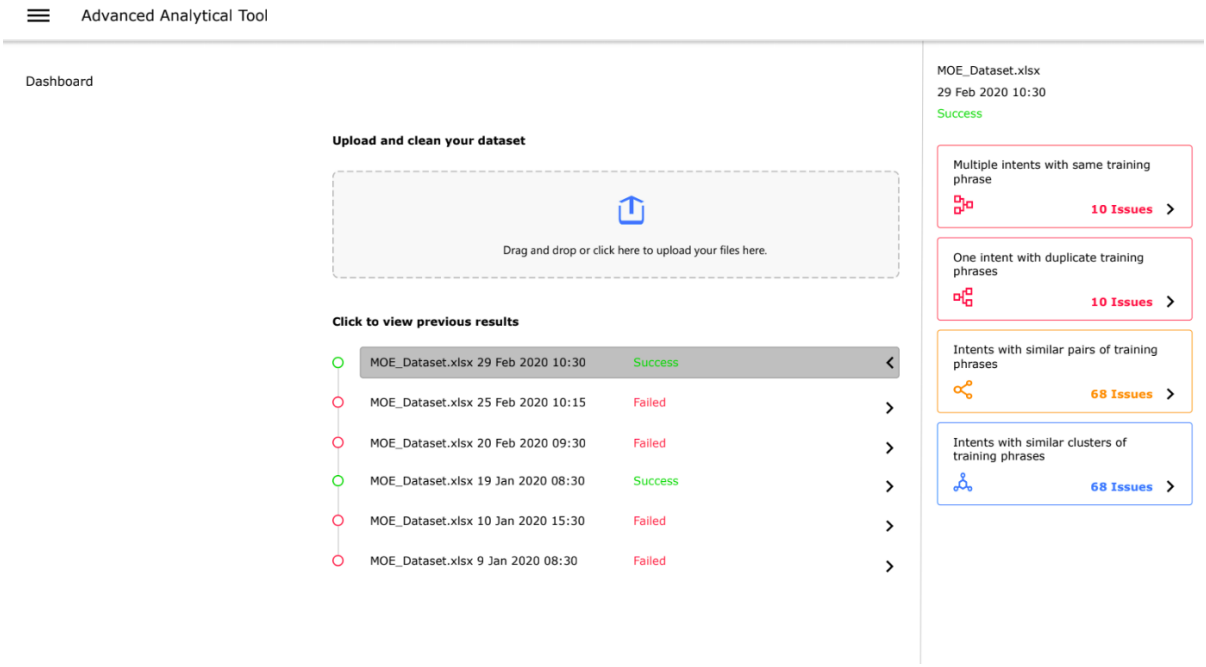

Figure 3: Mockups for Analytics Dashboard

My second main task was to setup and refine the NLP pipeline. The NLP pipeline is built to identify overlapping intents and ambiguous ones that affect the chatbot performances. I setup BERT and ALBERT to convert text into vectors of numbers (embeddings) that the computer can understand. The embeddings, theoretically, accounts for the semantic meaning. Cosine similarity and hierarchical clustering are performed on these embeddings to identify the problematic intents in the chatbot dataset.

As the chatbot dataset can be huge, I optimize one part of NLP pipeline so that the RAM usage

will not explode. I also provisioned a AWS EC2 Memory and GPU optimized instances to run my NLP pipeline.

The NLP pipeline helps to identify problematic intents in the training dataset. The source of this problematic intents is mainly a result of end users not understanding that a chatbot cannot handle ambiguous intents. Therefore, training phrases need to be specific. Secondly, end users may not have realized the intents already exist in the training dataset, and added new ones that results in a conflict. Table 2 shows an example of overlapping intents. My clustering tool in NLP pipeline grouped these set of training phrases together. It is obvious by looking at the training phrases alone, it is difficult to guess the intent. Similarly the chatbot would face the same difficulty that could have resulted in the chatbot's poor performance.

Table 2: Sample results of NLP pipeline

Training Phrases	**Intents**
Tax reference No Fxxx	How do I contact IRAS?
ref no Fxxx	GPF 58 What is the User ID? What is my tax reference number?
Tax reference no Gxxx	What is the User ID?

2.4 Training Assignments Completed in 3rd Month

One task I did was Topic Modeling. I use Gensim and Mallet to identify topics in positive and negative sentiments. These topic helped Product Owners understand the strength and weakness of current digital products for areas of improvements. A sample result of the topic modeling for negative sentiment is show in Table 3.

Topic modeling identifies a set of most representative keywords that could help Product Owners deduce the topic. I also find the most representative sentiment for each topic to provide context to these keywords. For example, Topic 1 in Table 3 suggests caching of user details so that user does not have to re-input the same details subsequently.

Another task I completed was to setup the backend code in Flask for the analytics dashboard. All the endpoints for the backend have been exposed. Sentry is setup to monitor for errors during

Table 3: Sample results of topic modeling

Topic	Keywords	Text
1	card, credit, detail, key, info, save, store, scan, regular, future	Re-input of credit card payment details each time. Maybe ca scan or take photo of credit card to capture details or save last transaction details.
2	card, detail, credit, key, remember, input, type, everytime, color, product	Remember my credit card number except my ccv number expiry date. At least I can input my card number w/o going to wallet downstairs to key in again.

runtime. PyTest was setup for unit and integration test. SwaggerHub was also setup for document the RestAPIs. RQ and Redis was also setup for the queuing system to run long computation tasks in the background. MongoDB was also setup to store results in the NLP pipeline. The backend code is also dockerized and deployed on AWS.

2.5 Training Assignments Completed in 4th Month

One task I completed was to deploy the application on Kubernetes using AWS EKS. The finished product on the staging environment is shown in Figure 4. There were some modications made to the UI to make the dashboard more user friendly and intuitive. The application is deployed on Kubernetes to orchestrate the various Dockers to run the application. I also setup the Kubernetes dashboard to visualize the different resources and services in my Kubernetes cluster. Kubernetes has also help me to scale up and down resources base on the traffic, and update and deploy new images.

Another tasks I completed was to setup the regression test using Docker Compose. This allows the integration and unit testing to be containerized so that system requirements and dependencies

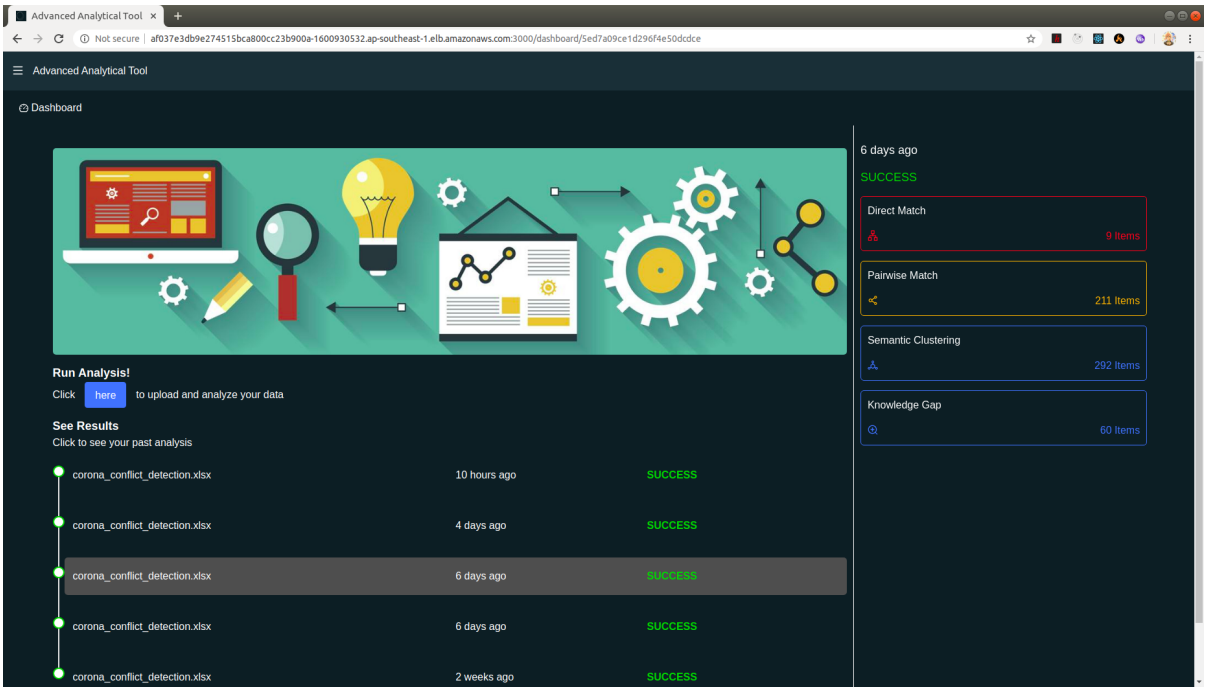

Figure 4: Dashboard on Staging Environemnt

are tested on the same environment as production's during continuous integration. To determine whether continuous integration is successful, the image for testing is built and run. The exit status of the image after the test is finish is listened to determine whether all tests passed or not.

2.6 Training Assignments Completed in 5th Month

One task I did was to develop the APIs to allow users to migrate the training dataset from FlexAnswer to DialogFlow. The way the data are stored in both platforms are different, and requires some data structures and algorithms, such as Breath First Search and Depth First Search, to group the training dataset into the correct parts of a conversation flow. Besides, there are many edge cases due to poor integrity of the dataset or missing values. Currently, the user first need to upload their DialogFlow chatbot's credentials on the dashboard, shown in 5, to later upload and migrate the data from FlexAnswer to DialogFlow.

Another tasks I completed was to setup logging on both the frontend and backend. I used the AWS CloudWatch to log the messages from both frontend and backend to. I found interested libraries, such as winston-cloudwatch for frontend and watchtower for backend to convinently setup a queue

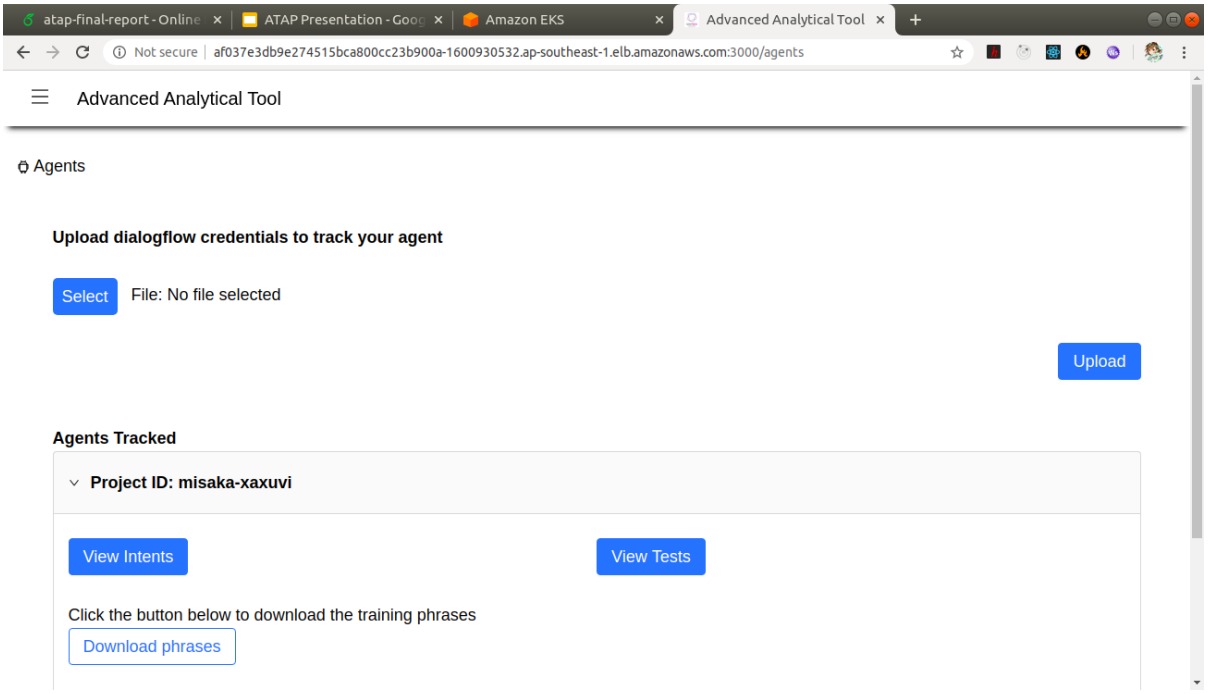

Figure 5: Agents Page in Dashboard

system to log the messages without encountering conflicts. The logging had helped one of the team member to debug the deployment error with Nginx which rejects headers with underscores.

2.7 Training Assignments Completed in 6th Month

One task I did was to build a scalable pipeline that provides the same analysis as the pipeline that I deployed to the Kubernetes cluster. PySpark allows the memory consumed to be distributed; therefore, the analysis can be scalable. I used SparkNLP to extract out the embeddings and performs KMeans with the distance metrics as cosine similarity. The scalable pipeline download the files from AWS S3, reads the dataset, performs similar analyses, and then uploads the files back to AWS S3. The only difference is that it is using PySpark for scalability.

Another tasks I completed was to implement APIs to perform CRUD on the intents, training phrases, entities, and responses. Last month, I made a similar attempt for the intents and training phrases; however, the approach is not production ready and stable. The current changes made is to mark items deleted with a boolean flag instead of actually deleting the item. Secondly, the training phrases are stored a string without its entities. I made the changes to the implementation

to support storage of the training phrases with entities.

3 Knowledge And Experience Gained

3.1 Technical Knowledge Gained From Assignments

The first technical knowledge I gained was learning about NLP. I had the opportunity to learn different unsupervised clustering techniques, such as HDBSCAN, Agglomerative, and DBSCAN. I also had hands-on experience with Tensorflow and PyTorch to use BERT and ALBERT to convert text into embeddings. Another opportunity I had was to do Topic Modeling for positive and negative sentiments with one of the datasets using Gensim and Mallet. I later learned about using TextRank as an alternatives for Topic Modeling.

The second technical knowledge I gained was developing the web-based platform from scratch. I also had to do mock-ups and conduct user testing to check whether my designs would be intuitive for end users. Having taken CS3216, I wanted to apply the technologies and practices I learned from my seniors when I worked on projects with them. I am current using ReactJS with Typescript, Flask, and MongoDB. The application is deployed over Docker. I also integrated a queue based system using RQ to managed processes that requires long computation time in the background.

The third technical knowledge I gained was deploying the application over Docker and Kubernetes (AWS EKS) on AWS. I also setup my unit and regression tests to use Docker Compose so that the testing is also containerized. This allows continuous integration to done before deployments without having to worry about system requirements. Having the opportunity to deploy the application on Kubernetes, I learned that Kubernetes allows the application to scale easily, update and deploy new images easily, and monitor the health of the application easily through Kubernetes dashboard. I also learned about AWS Fargate to manage the containers and instances type even though I did not use it for deployment.

The fourth technical knowledge I gained is using PySpark to build scalable models for the pipeline. As the training dataset can have over 400,000 utterances and the RAM usage is exponential to the number of utterances, performing clustering on the 400,000 utterances can take up more than 200GB RAM. While I converted my original pipeline using Pandas, Scikit Learn and PyTorch to PySpark, I learned to about the capabilities of PySpark to distribute the memory across multiple platforms, making the pipeline highly scalable despite some memory and computation overheads. I currently using PySpark to perform text analytics and embeddings extraction, giving the same

pipeline as before but scalable to large datasets. The PySpark code is deployed on AWS Glue which provides managed ETL services, solving the pain of not having enough RAM.

3.2 Organizational/Industry Experience Gained From Assignments

The first industrial experience I gained was sharing my NLP results with government agencies. The very first presentation I gave was to MOE, but that was a last minute notice and I did not have any slides prepared. During the meeting, I could tell the audience could not follow. After that incident, I prepared a PowerPoint template to share my results that is easy to understand for end users. When I finally presented my results to MOE and IRAS again, there was a huge difference and the audience were asking questions that showed they understood my sharing. Most importantly, the government agencies know how to use the CSV files to clean their chatbot's dataset.

The second industrial experience I gained was working with the team. My team has daily stand-ups which I was unused to at first, as I was clueless the first few weeks and did not know how to report my activities in NLP. After asking around and observing how others report, I was able to. My team also uses JIRA to track issues and also uses it for sprint planning. It took me about a week to adjust because I do not use such platform often in school. The most exciting part was emergency request to develop a chatbot within 2 days to disseminate information about masks due to the Coronavirus situation in Singapore. I first-hand saw how a single developer was able to build it in 2 days. Even though I training the chatbot through DialogFlow that 2 days, I helped out with redesigning the chatbot interface, scraping MOH web contents to populate the statistics into the database, and automating the regression testing for the training dataset.

The third industrial experience I gained was understanding the terminologies that government agencies used. In my first team meeting, the Director was present to supervise the team's progress. During the team meeting, there were many acronyms thrown that left me confused. I was not familiar with many of the government agencies and little did I know the purpose of each of these agencies. However, I was able to follow the conversations in subsequent meetings as I became more familiar with the terminologies.

3.3 Areas of Applicability of Knowledge And Experienced Gained

NLP is a niche field in Machine Learning and Deep Learning, but is growing rapidly. NLP has directly applications to any projects related to developing chatbot and understanding consumers' sentiments. The concepts and skills in Machine Learning and Deep Knowledge, however, is applicable to other fields, such as Computer Vision. As I hoped to develop a talking robot that can conversational capabilities of a human one day, I believe the NLP skills I gained at GovTech would help me in future.

4 Conclusion

4.1 Summary Of Work Completed And Training Received

The training I received was learning about the project's background and the skills in NLP.
One of the major work I completed is setting up the NLP pipeline that can easily integrate into the backend code.
The second major work I completed is the setting up the backend endpoints to fetch data from database.
The third major work I did is deploying the application over Kubernetes and setting up logging in the system.
The fourth major work I did is setting up PySpark to run text analytics and embeddings extraction on large datasets.
Other works I have done include topic modeling, web scraping, user testing with mockups, queue system for pipeline, sharing NLP results, redesigning chatbot interface, automating regression testing, and training chatbot with DialogFlow.

4.2 Problems Faced

One problem I am investigating is setting up the PySpark to run on AWS Glue and have it play nicely with AWS S3, AWS Step, and AWS Lambda
The second issue I need to solve is to learn and use terraform to managed AWS services.

4.3 Assessment Of Training Experience And Concluding Remarks

I enjoyed learning about NLP. In fact, I am learning how to use NLP for foreign languages, since I am also learning a foreign language currently. I hope to have opportunities to learn Natural Language Understanding (NLU) in chatbot if the project caters for it. Besides my current team, I would be joining another team to work on Robotics and Artificial Intelligence. I cannot wait to learn more about Machine Learning and Deep Learning in the other team.

References

[1] L. Hio, "Parliament: Bills to merge ida and mda, and to form new govtech agency, passed." https://www.straitstimes.com/singapore/parliament-bills-to-merge-ida-and-mda-and-to-form-new-govtech-agency-passed, Aug 2016. Accessed on 2020-03-30.

[2] G. T. Agency, "Digital government transformation." https://www.tech.gov.sg/digital-government-transformation/, Mar 2020. Accessed on 2020-03-30.

[3] G. T. Agency, "Digital government blueprint." https://www.tech.gov.sg/digital-government-blueprint/, Mar 2020. Accessed on 2020-03-30.

[4] G. T. Agency, "Students and graduates." https://www.tech.gov.sg/careers/students-and-graduates/, Mar 2020. Accessed on 2020-03-30.

[5] C. S. C. Singapore, "Learn.gov.sg." https://play.google.com/store/apps/details?id=cxs.cscollege.learn&hl=en_SG, Feb 2020. Accessed on 2020-03-30.

[6] S. Nation and D. G. Office, "Moments of life." https://www.smartnation.sg/what-is-smart-nation/initiatives/Digital-Government-Services/moments-of-life, Sep 2019. Accessed on 2020-03-30.