COSC 310 - Assignment 2

Ian Heales - 40183402
Thomas Van De Crommenacker - 33970138
Andres Escobedo - 44365154

Python Chat Bot

Project Description

We are creating a chat-bot in Python through machine learning. Our chat-bot plays the role of a friend to the user. The project's current iteration uses the Spacy library for basic natural language processing and the Keras library to create and train a basic End to End Network (Sukhbaatar, Sclam, Weston, & Fergus, 2015) used to predict answers based on the intent and actual user input, more or less through phrase matching based on our data set. The model is trained on a Dialogue Flow data set. Our team repository is hosted on GitHub.

SDLC

We chose the Spiral Rapid Prototyping development model for this project because the requirements of this project will change as we learn about NLP and chat-bots. While chat-bots are not a new technology building them is new to us. Spiral will allow us to create working prototypes early on through the iterations to get hands on, review progress, and learn about the technology. In addition, spiral works best for short iterations known as loops, which is ideal for our intention of using our weekly lab period for the reviewing and planning stages. This was done due to our ever changing understanding of NLP and of the requirements of a functional chat-bot. Spiral allows us to dynamically identify objectives, alternatives and constraints of our current prototypes. The team members will take on both the role of the development team as well as customer or product owner. Additionally our team formation follows the democratic team approach. This is mainly due to the team consisting of only 3 individuals, all of equal technical ability.

SDLC Phases

Several Iterations of the Spiral SDLC - Denoted by number with basic descriptions.

Phase 1 : Determine objectives, alternatives and constraints.

1.
    1.1 Choose between machine learning and pattern matching.
    1.2 Consider the time frame of the project.
    1.3 Choose a programming language and libraries.
    1.4 Determine programming experience of group members.
    1.5 Create a Gantt chart.
2.
    2.1 Check out Ludwig and Rasa.
    2.2 Add to data-set
    2.3 Implement Intents sensing.

COSC 310 - Assignment 2

Ian Heales - 40183402
Thomas Van De Crommenacker - 33970138
Andres Escobedo - 44365154

Phase 2: Evaluate alternatives, identify, and resolve risks

1.
    1.1 Learn about machine learning.
    1.2 Determine role of the chat-bot.
    1.3 Find a data-set tailored towards conversation.
    1.4 Determine if the data-set matches the assigned role.
2.
    2.1 Re-evaluate if machine learning is the correct choice for this assignment.
    2.2 Test different methods of machine learning (Ludwig / Rasa).
3.
    3.1 Use Ludwig or Rasa for intents sensing.
    3.4 Determine if we can train the current model to sense intents.

Phase 3: Develop next-level product

1.
    1.1 Create simple prototypes using libraries.
    1.2 Implement sentence intent sensing.
2.
    2.1 Improve on prototype such that it can produce more complex answers.
    2.2 Ensure prototype does not break when incorrect input is entered.
3.
    3.1 Create a simple GUI.
    3.2 Retool data-set.
    3.3 Use current model to filter intents.
    3.4 Working System.

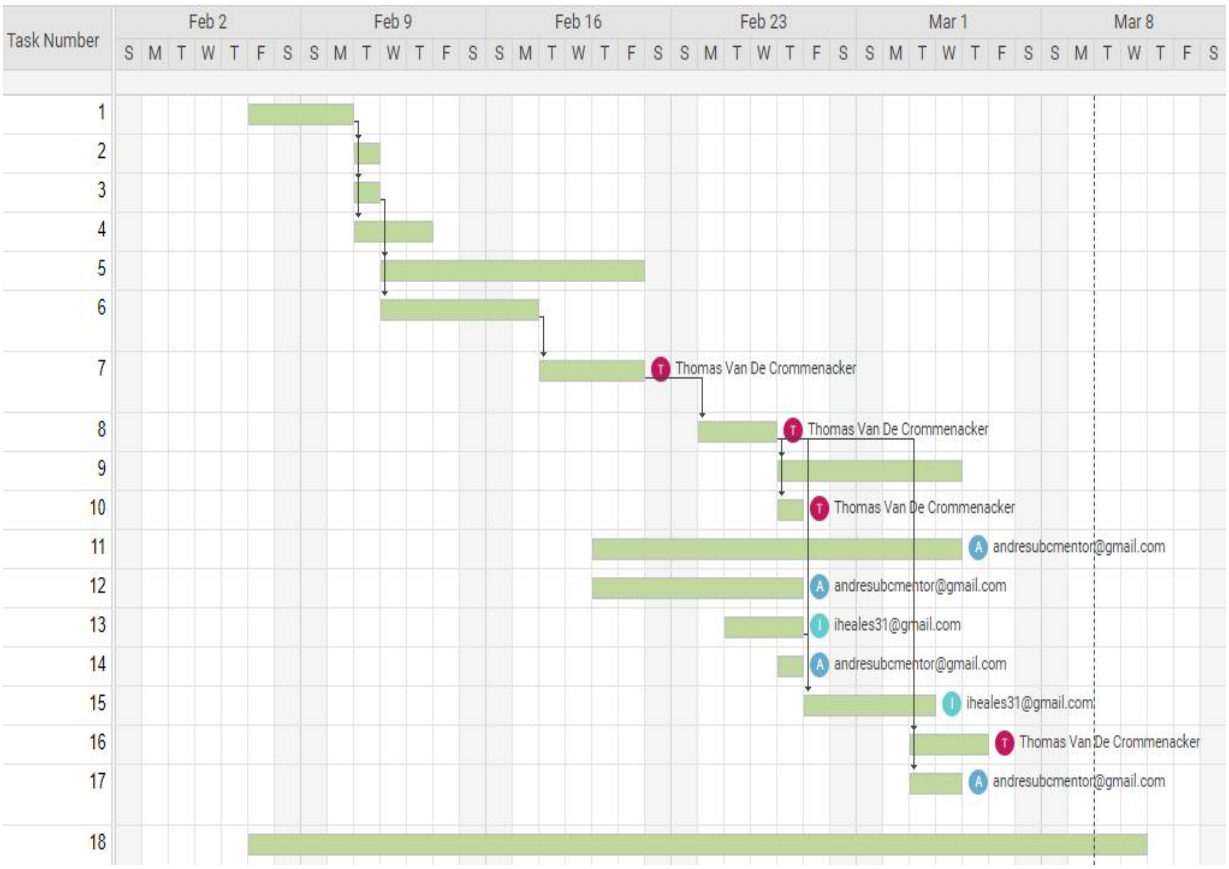
Phase 4: Plan next phase

1.
    1.1 Expand chatbot conversational options.
    1.2 Find new datasets.
    1.3 Find other libraries.
2.
    2.1 Expand on keras prototype.
    2.2 Evaluate Ludwig and Rasa capabilities.
3.
    3.1 Allow for chat-bot to chat-bot conversations using sockets.
    3.2 Improve GUI.
    3.3 More NLP features. (POS, Entity recognition etc)
    3.4 Potentially hosting it somewhere.

COSC 310 - Assignment 2

Ian Heales - 40183402
Thomas Van De Crommenacker - 33970138
Andres Escobedo - 44365154

Gantt Chart



Task Number	Task Name	Duration	Start	Finish	Pre...	As... To	Comments
1	Decide on Language	2d	02/07/20	02/10/20			
2	Create GitHub Repository	1d	02/11/20	02/11/20	1		
3	Brainstorm Meeting	1d	02/11/20	02/11/20	1		
4	Research similar examples	3d	02/11/20	02/13/20	1		
5	Look for Training Sets	8d	02/12/20	02/21/20	3		
6	Research - Python Libaries / ML?	4d	02/12/20	02/17/20	3		Several hours over the course of February 17-20 doing research and creating notes for future reference. Settled on Spacy or keras for ML approach.
7	Wiki QA bot prototype	4d	02/18/20	02/21/20	6	T	Created a chat bot which can only answer questions from a wikipedia data set. For this, you need to specify the subject and then ask the question exactly as it is written in the data set.
8	Keras Bot Prototype	3d	02/24/20	02/26/20	7	T	This is the prototype which stemmed into our final chat bot
9	Refactor Keras Bot	5d	02/27/20	03/04/20	8		
10	Bot Only Script	1d	02/27/20	02/27/20	8	T	Created a script which only utilizes the model to interact with the chat bot. You cannot train the model with this script.
11	Create Gantt Chart	10d	02/20/20	03/04/20		A	The Gantt chart was made early on, but altered as our plans and requirements evolved.
12	Research Rasa NLU Library	6d	02/20/20	02/27/20		A	A library which utilizes intents. the libraries did not work with our versions of Python.
13	Create GUI	3d	02/25/20	02/27/20		I	Created a simple GUI to have a chat log with the chat bot.
14	Research Uber's Ludwig	1d	02/27/20	02/27/20		A	Attempted to use Uber's Ludwig but the libraries didn't seem to work.
15	Incorporate GUI	3d	02/28/20	03/03/20	13, 8	I	
16	Intent Filter	3d	03/03/20	03/05/20	8FS +	T	No longer need to specify a topic before asking a question/saying something to the chat bot.
17	Answering Unknown vocab	2d	03/03/20	03/04/20	8FS +	A	Allowing the chat bot to answer that it does not know what the user is saying. The chat bot picks from many random statements which acknowledge that the chat bot doesn't know how to answer.
18	Documentation in a Google Doc	24d	02/07/20	03/11/20			Everyone continuously contributed to the documentation in an online google doc throughout the assignment duration.

COSC 310 - Assignment 2
Ian Heales - 40183402
Thomas Van De Crommenacker - 33970138
Andres Escobedo - 44365154

Work Breakdown Structure

Top Level WBS

Numbering based on suggested WBS in lecture slides. Not directly related to SDLC phases.

Phase No.	Phase Name
1.0	**Initiation**
2.0	**System Study**
3.0	**Detailed Design**
4.0	**Development**
5.0	**Implementation**
6.0	**Project Completion**

Breakdown of Individual Phases WBS

Phase 1: Initiation

Task No.	Task Title:	Estimated Hours:	Actual Hours:	Assigned to:
1.1	Create GitHub Repository	10 Min	10 Min	Thom
1.2	Choose a programming language - Research.	1 Hrs	1 Hrs	All
Results:	Python was chosen for its potential ease and libraries.			
1.3	Choose between Machine Learning and other techniques - Research.	5 Hrs	10 Hrs	All
Results:	Machine Learning was chosen for challenge and potential.			
1.5	Create a Gantt Chart	4 Hrs	5 Hrs	Andres
Results:	Andres used smartsheet to create the Gantt chart and added multiple entries which could be edited easily by other team members since it is online based and can be shared.			
1.6	Create official document plan with current information	2 Hrs	4 Hrs	All

Phase 2: System Study

Task No.	Task Title:	Estimated Time:	Actual Time:	Assigned to:
2.1	Choose a role for chat-bot	30 min.	1 Hrs	All
Results:	The role will be decided by the data-sets that we can find.			
2.2	Research and choose appropriate data-set	5 Hrs	5 Hrs	All
Results:	Several data sets were found. The first to be used will be a Wiki Facts data-set. Another data-set from Dialogue Flow looks more promising. Role of chat-bot will be a friend.			

2.3	Research Machine Learning Libraries and Techniques using Python	5 Hrs	10 Hrs	Thom
Results:	Thom researched different Python libraries for NLP and ML capabilities, settled on Keras and Spacy for the first prototypes			
2.4	Review current prototype, progress and SDLC phase - Update Documentation	2 Hrs	2 Hrs	All

Phase 3: Detailed Design

Task No.	Task Title:	Estimated Time:	Actual Time:	Assigned to:
3.1	Determine the type of ANN / RNN to be used and how.	5 Hrs.	8 Hrs	Thom
Results:	Settled on End to End Network based on online course and paper research uncovered. Basic model takes in two inputs and uses it to predict the output. Current application does not take advantage of its potential but predicts answers well. See figure 1.			
3.2	Handle unknown inputs	2 Hrs	2 Hrs	Andres
Results:	Chat bot will respond with randomized responses that suggest it doesn't know how to respond whenever it is given an input which contains no known vocabulary.			
3.3	Add more conversational topics to chat-bot.	2 Hrs	1 Hr	Thom
Results:	Thom added more questions to flesh out the chat-bot's role.			
3.4	Review current prototype, progress and SDLC phase - Update Documentation	2 Hrs	2 Hrs	All

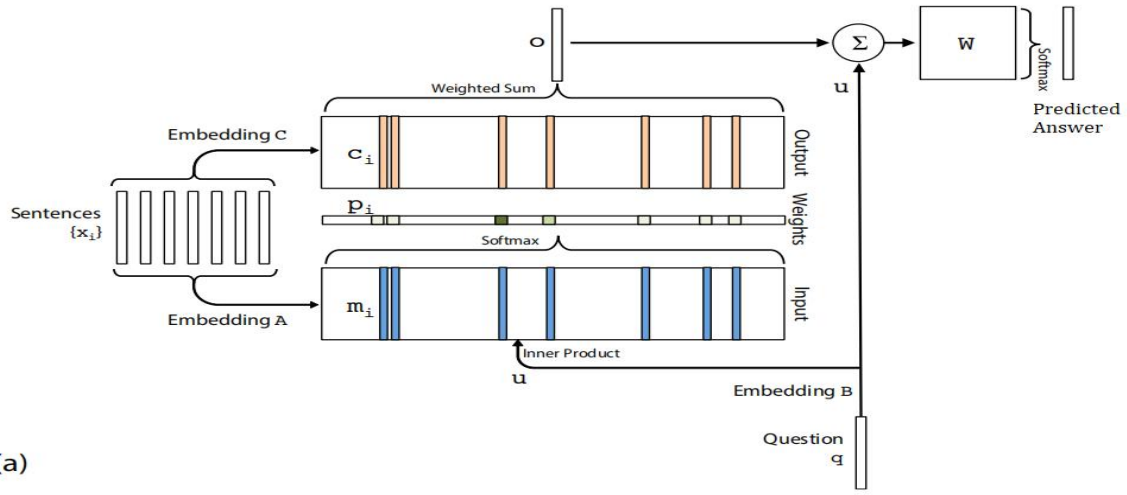


Figure 1 : Single layer of the End-to-End model.(Sukhbaatar, Sclam, Weston, & Fergus, 2015)

Figure 1 is of a single layer of the End to End network we plan on implementing based on the *End to End Networks* paper. It essentially takes in three inputs during training. The "sentences" here will be the intent and question which will be split, vectorized and embedded using an embedding matrix. Which is essentially just projecting the large matrix of

sentence vectors into a more meaningful “space”. The third input is the answer it needs to predict. The vectorization of the sentences works as follows. We will have to create a set of words, meaning no duplicates, and give each word an index. That way the sentences can be represented as vectors using those indexes. Due to the different sizes of sentences, we’ll find the maximum length and pad the others with zeros. The internal mathematics of how the system is trained to find the answer can be found in the paper. A very basic “yes and no” interpretation of the model itself will be implemented with the help of Pierian Data’s lecture resource and then augmented and built upon to create our desired functionality. We plan on essentially using two layers or versions of the model, where the first one is trained on solely the questions to predict the intent, and the second layer for predicting the answer based on the predicted intent and the question. For the output the model is trained to output the particular index of the answer out of the pool of answers. See the data flow diagram for a more detailed explanation.

Phase 4: Development

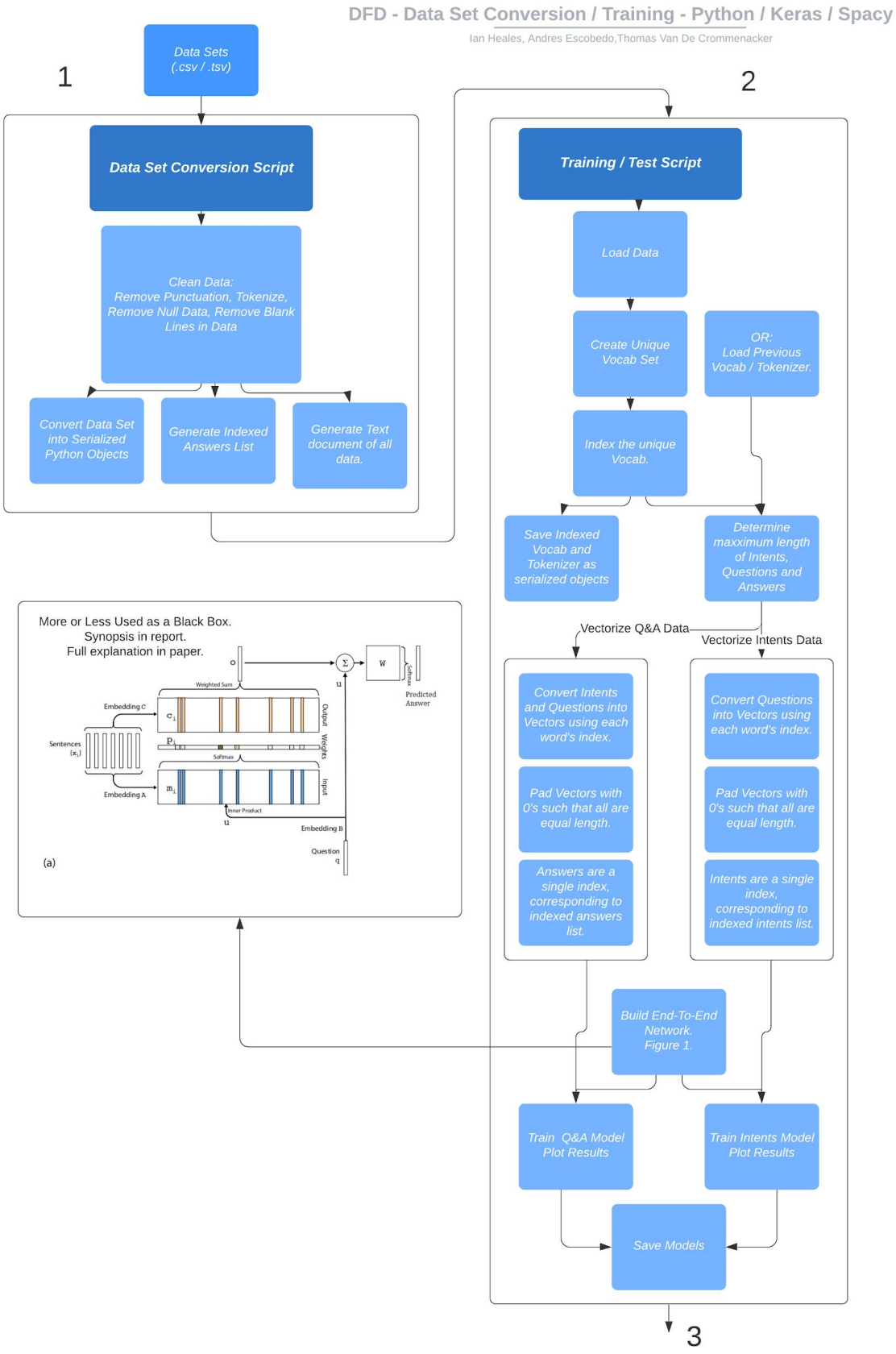
Task No.	Task Title:	Estimated Time:	Actual Time:	Assigned to:
4.1	Implement Basic Prototype based on lecture resources.	4 Hrs	4 Hrs	Thom
Results:	A basic yes/no prototype was created (keras_prototype_v1). The model’s structure and function is well understood.			
4.2	Expand on the basic prototype to be able to use more complicated data-sets and generate answers. Most of the time.. **(prototype v2 - v5)**	2 Hrs	4 Hrs	Thom
Result:	Chat bot can now produce more complicated answers than yes or no.			
4.3	Implement first GUI	1 Hr	2 Hrs	Ian
4.4	Find or create a more suitable data-set for the chat-bot than the Wiki Facts data set.	1 Hr	2 Hrs	Andres
Results:	After Research Andres found a more suitable data set from dialogue Flow to serve as the basis for the chat bot.			
4.6	Figure out intent filter and fine tune current working model.	2 Hrs	3 Hrs	Thom
Result:	Another layer / model was trained to use the question as inputs and produce the appropriate intent to be used in the next layer.			
4.7	Review current prototype, progress and SDLC phase - Update Documentation	2 Hrs	2 Hrs	All
4.7.1				

Data Flow Diagram

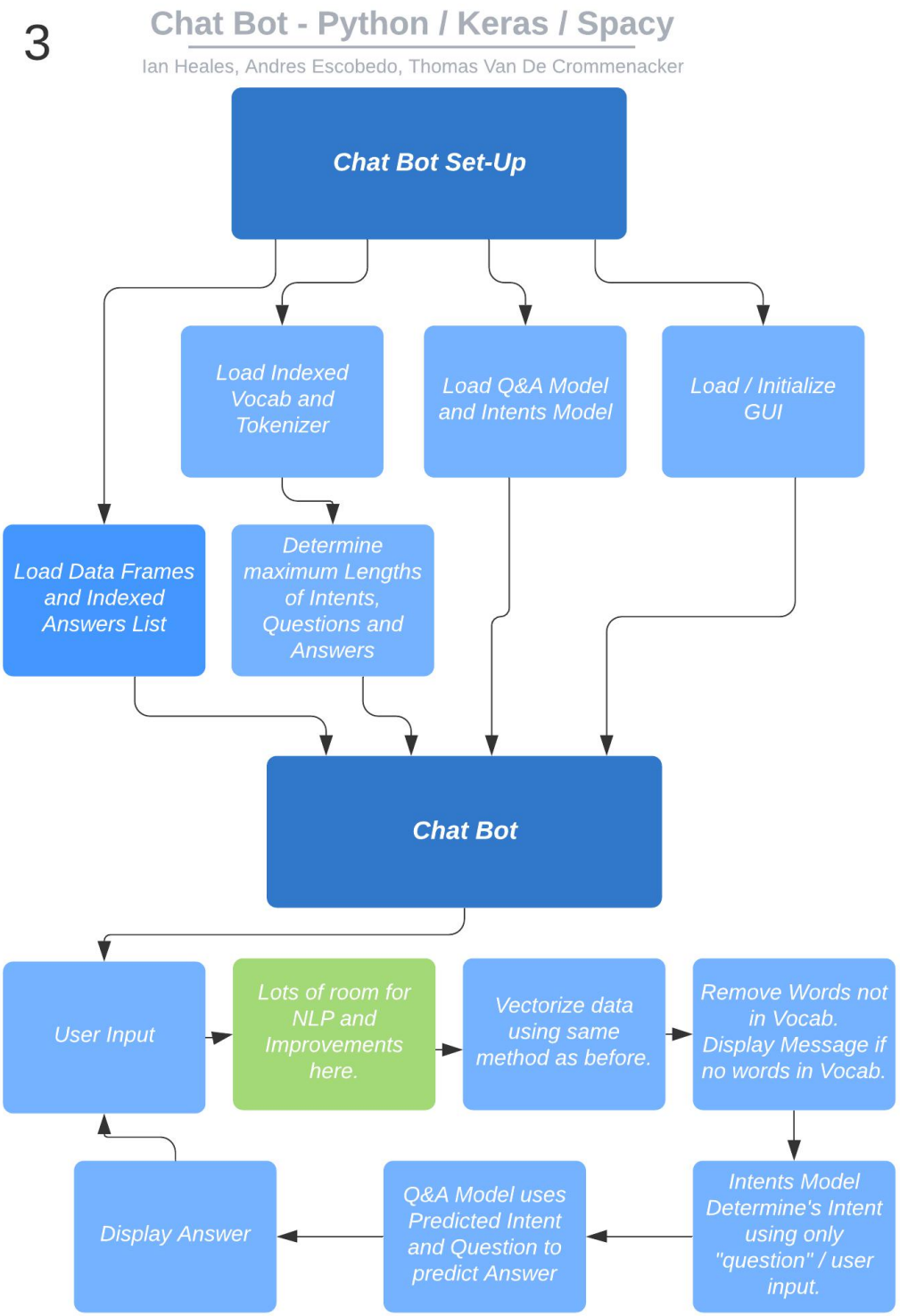
Due to the nature and relatively short length of our scripts the code exists in only 3 separate python script files. The conversion script is used to convert and serialize the data sets into the appropriate Python objects to be used during training. The second is a script that loads in the objects and appropriate files from the conversion script and trains the model, this script also has the functionality of not training and simply loading and chatting. The third script is the deliverable or distributable file as it contains solely the chatting functionality.

COSC 310 - Assignment 2

Ian Heales - 40183402
Thomas Van De Crommenacker - 33970138
Andres Escobedo - 44365154



Ian Heales - 40183402
Thomas Van De Crommenacker - 33970138
Andres Escobedo - 44365154



Limitations

- Chat Bot does not currently have a specific topic besides small talk and being a bot.
- Going off script can produce incorrect output frequently.
- Chat Bot Does not remember previous chat data and therefore does not learn from the user input.
- Chat Bot does not have a very large data-set.

Sample Output

Show a 30-turn dialogue. Show 2 short dialogues where the agent is not able to handle the conversation properly.

30 Turns Dialogue - Referencing data-set.
User: Hey!
Chatbot: hi there friend !
User: how are you doing?
Chatbot: lovely thanks .
User: want to grab something to eat?
Chatbot: i had a byte just now .
User: That sucks.
Chatbot: cancelled ! just tell me what you 'd like me to do next .
User: wait are you a robot?
Chatbot: indeed i am . i 'll be here whenever you need me .
User: WHAT
Chatbot: your perception . my reality .
User: You must be a human.
Chatbot: thanks i try .
User: could i gibe you a hug?
Chatbot: hug it out . you 'll feel better afterwards .
User: thank you
Chatbot: thanks ! the feeling is mutual .
User: ...
Chatbot: I'm not sure what that means.
User: are you still there?
Chatbot: right where you left me .
User: where do you live?
Chatbot: right here in your device . whenever you need me .
User: where do you come from?
Chatbot: some call it cyberspace but that sounds cooler than it is .
User: woah
Chatbot: wow indeed !
User: do you have a hobby?

Chatbot: i should get one . it 's all work and no play lately .
User: haha what do you do for work then?
Chatbot: great !
User: that is not right
Chatbot: of course i am .
User: no
Chatbot: okay then .
User: you are fired
Chatbot: oh do n't give up on me !
User: too late.
Chatbot: likewise !
User: seriously it is time to fire you
Chatbot: thanks ! the feeling is mutual .
User: you're really funny
Chatbot: thanks .
User: you're so boring
Chatbot: i 'm sorry you think so . we can talk about something more interesting .
User: alright well tell me about your personality
Chatbot: just think of me as the ace up your sleeve .
User: quite a bold statement
Chatbot: okay then .
User: what is your age?
Chatbot: i 'm a relatively new bot but i 'm wise beyond my years .
User: I disagree
Chatbot: i understand .
User: gah. I am so overworked
Chatbot: you should get some shuteye . you 'll feel refreshed .
User: I promise to come back
Chatbot: hooray !
User: I'd be happy to see you again.
Chatbot: i certainly hope so . i 'm always right here whenever you need me .
User: thanks i will, I should go to bed
Chatbot: pleasant dreams . see you soon .

Dialogue showing shortcomings version 1:
User: What is up!
Chatbot: living the dream .
User: that is nice, can I ask you a question?
Chatbot: i 'm not programmed for that exact question . try asking another way ?
User: Where was isaac newton born?
Chatbot: i wish i knew where .
User: That actually kind of works
Chatbot: okay then .
User: what time is it?
Chatbot: cancelled . waiting for more commands .

User: if i ask you anything random, what is your response?
Chatbot: okay then .
User: is okay then just your default response to questions you dont know?
Chatbot: terrific !
User: haha very clever
Chatbot: thank you . i try my best .
User: Not bad, you are getting better
Chatbot: terrific !
User: but you are annoying
Chatbot: sorry to come across that way .
User: what is a red black tree?
Chatbot: okay then .
User: what is the faraday effect?
Chatbot: okay then .
User: what did james watt invent?
Chatbot: okay then .

Dialogue showing shortcomings version 2:
User: do you know anything about software engineering?
Chatbot: okay then .
User: I thought so.
Chatbot: i see .
User: We'll get you working yet. More testing!
Chatbot: i encourage you to test me often . that helps my developers improve my performance .

Meeting Minutes

Participants for all meeting were Thomas Van De Crommenacker, Ian Heales and Andy Escobedo.
February 1st

- All - Deciding on a language to create the chatbot.

February 11th

- Ian - SDLC justification.
- Ian - Set up google doc.
- Thom - Explored options (pattern matching vs. machine learning vs both?).
- Andy - Gantt chart setup.
- All - Exploring options for creating a chat bot (4 hours).

February 19th

- Thom - Researched spaCyand NLTK.

February 23rd

- Thom - Simple chat bot.
- All - Look up alternate datasets.

February 25th

- Ian - Simple GUI using TKinter.
- Thom - New answer set, updated chatbot.

- Andy - Sentence intent sensing.
- Andy - Found dialog flow dataset (Dialogflow Prebuilt Agents GoogleSheet).
- Andy - Explored the use of Rasa.

February 27th

- Ian and Andy - Explored if Ludwig would be suitable for training the chatbot.

February 29th

- Thom - Updated README file.

March 3rd

- Thom - Refractored chatbot.
- Andy - Unknown/empty question handling.
- All - Project Report.

March 10th

- Andy - Found new dataset in YAML format. Created conversion script so that we can use datasets in these formats
- Thom - Added more questions to the dataset.
- Thom - Trained chatbot on new dataset.
- All - Finalized project report.

Citations

code-free deep learning toolbox. (n.d.). Retrieved from https://uber.github.io/ludwig/

Dialogflow. (n.d.). Retrieved from https://dialogflow.com/

Keras: The Python Deep Learning library. (n.d.). Retrieved from https://keras.io/

Natural Language Toolkit. (n.d.). Retrieved from https://www.nltk.org/

Open source conversational AI. (n.d.). Retrieved from https://rasa.com/

spaCy · Industrial-strength Natural Language Processing in Python. (n.d.). Retrieved from https://spacy.io/

Sukhbaatar, S., Sclam, A., Weston, J., & Fergus, R. (2015, March 31). End-To-End Memory Networks. Retrieved March 10, 2020, from https://arxiv.org/abs/1503.08895