Challenges Associated with Architecting a Voice User Interface (VUI) and Text User Interface (TUI) for Natural Language Understanding (NLU) / Processing (NLP)

Aaron A. Gauthier

Natural Language Processing

CCAS – Data Science Department

The George Washington University

December 15, 2019

Challenges Associated with Architecting a Voice User Interface (VUI) and Text User Interface (TUI) for Natural Language Understanding (NLU) / Processing (NLP)

Introduction

Problem. Ambient computing is the future and is already all around us. Simply put ambient computing is the collection of devices we use at home and work. They essentially become extensions of each other and offer us an overall seamless user experience. It's the combination of hardware, software, user experience and machine/human interaction and of course learning. Prototyping the user experience can be hard – particularly with VUIs. Getting the VUI correct can be challenging for a multitude of reasons. If you get the design wrong – then your user will have a horrible experience and more than likely won't use your skill again! Learning how to architect this correctly is paramount. Some of the toughest challenges to resolve will be error handling, misspellings, mispronunciations (including accents). This project shall include but not limited to the utilization of out of the box functionality of AWS including Lambda, Lex, Alexa as well as Twilio. AWS functionality will be integrated with Twilio

Motivation. The purpose of this project was to learn how to build a Voice User Interface (VUI) / Test User Interface (TUI) utilizing AWS Lambda, Lex, Alexa, and Twilio. I have read multiple blogs and watched many YouTube videos about how complex architecting a VUI can be – so I wanted to give it a try. I wanted to see for myself just how complicated it can be to architect a VUI. Additionally, the future systems will include VUI – there is no way around it. Keeping abreast of technological trends is very important especially when the future involves VUI's. There is a lot of interest in the public and private sectors for incorporating Alexa like

functionality into future intelligent systems to solve business problems and make systems overall more user friendly and by default more intuitive. Alexa like skillsets have broad applications in today's modern world. Learning how to architect, design and program such skillsets are a very valuable skill in today's world.

Domain Knowledge. While reading about VUIs, AWS functionality and researching various aspects of Twilio, Lambda, Alexa and Lex – I have accumulated enough understanding of this problem set. I have learned to do this correctly one really needs to prototype with these technologies often – slowly increasing the level of complexity to completely learn about the technology, its limitations and how to create a methodology that works every time in architecting these solutions. You must understand prototyping, design methodology, nuances of language ambiguity, detailed planning to ensure you lead your user to their desired outcome all the while incorporating personality and ease of use. Not an easy task – but one that comes with experience. Additionally, constructing wire diagrams and leveraging the use of Adobe XD's Alexa Skillset for prototyping would significantly help in planning out a VUI.

Proposed Methods

AWS Lambda. Lambda allows you to run code without provisioning servers and zero administration. Lambda automatically scales your code with high availability. Code can be automatically configured with a "trigger" from other AWS services or it can be called directly through an application. Lambda is the preferred solution for use with Lex and Alexa. I will leverage Lambda in the background as part of the overall architecture.

AWS Lex. Lex is a service used for building conversational interfaces utilizing text and voice. Lex does have advanced features such as automatic speech recognition (ASR) and natural language understanding (NLU) for recognizing the intent of speech or for converting speech into text. These deep learning capabilities that are available with Alexa are available with Lex – however we did not have time to explore all the deep learning capability nor how to leverage it. This was strictly a very basic prototype. Lex was utilized as part of the architecture. This is an area of exploration for future work.

AWS Alexa. Alexa is a voice service available on AWS. It's available on more than 100 million devices and third-party device manufacturers. You can build more intuitive voice experiences for consumers to interact with technology. AWS offer's a collection of tools, API's, templates and documentation to assist in architecting Alexa solutions. This project tried to integrate Alexa into the landscape but to no avail. Currently – we are uncertain of how to do this or if Lex already does this for us. This may be a redundant part of the architecture – but more research will have to be done in order to verify it.

Twilio. Twilio is a developer's platform that allows communications through leveraging APIs for a multitude of reasons. You can leverage the platform to incorporate voice, text and video integration within user applications. One advantage to Twilio is that it manages messages behind the platform so it's seamless to the developer and user – meaning if you are message a T-Mobile, Sprint, AT&T and Verizon customer, Twilio takes care of the integration and different charges, data formats associated with these mobile carriers. Twilio is utilized in this project as the TUI which is very similar to that of a VUI which is why it was also chosen as part of the integration effort.

Results, Application and Challenges

This was a pizza order application that was relatively simple. I utilized the same workflow on a popular franchise website. This was leveraged because they are a major franchise and therefore must have the workflow worked out – which allowed for some savings. The Mr. Pizza application though relatively basic was quite complex to architect and anticipate the dynamics of a conversation. The charts in the presentation give a good representation of the conversational framework and workflows. The charts are to be read from top to bottom and represents that graph from left-most to the right most area of the graph. It would have been way to small had a picture be taken and inserted. Webgraphviz was utilized to create these diagrams. I like the Graphviz product and plan on leveraging it in the future to wireframe future solutions. It's easier to use than a power point slide.

The application functioned as designed – feedback from limited users were to utilize the AI portions of the technology more, as well as insert comment cards instead of a straight texting application as well as work on the pizza ordering bots' personality more. The app worked and functioned well – people were able to utilize it immediately. The challenges of designing an application of this magnitude and scale for a commercial entity would be that of truly understanding all the requirements including functional, technical, and UI/UX. In order to do this correctly it would take a few months and lots of wireframing and diagramming prior to building the actual application with the user community. Thinking through all the variations within a conversation was the hardest – anticipating questions, answers, conversational flow including repeating menu options are all challenges that were not fully and deeply solved. Much of this can possibly be solved through the use of deep learning and some of it is plainly a

technical challenge as well as a timing issue of how often to ask for some to repeat a question, when to terminate the conversation, and how and when to run through the list of choices if needed. It gets difficult for a food ordering application like a pizza when you have lots of choices. I also learned that accents, and pronunciation is very important as part of this solution. Those that are not native English speakers may have a difficult time pronouncing some words and hence Alexa having problems understanding which may add to frustration. The smart incorporation of multiple languages and being able to switch back and forth between languages is integral in leveraging such a solution but also adds exponential complexity into such an application or system.

Lastly for fun an Alexa class quiz was introduced to show what the art of the possible was with Alexa and to interact with it. Many of the challenges described above quickly came to light with Alexa. In utilizing Alexa, Lex and Twilio you have multiple options available to you including the use of templates as well as coding – including switching back and forth between templates and coding. When you first start with Alexa, Lex and Twilio utilizing the templates is a good way to start off – however as complexity gets introduced then it makes more sense to switch over to coding the solution. Certain aspects of Alexa, Lex and Twilio you only have the option of utilizing a template for various configurations. You need to utilize both, and I believe through this project have found the balance – however more time and experience will be required to confirm this.

Conclusions

This project has confirmed that in many ways implementing an Alexa, Lex and Twilio application can be quite easy if it's a one way highly structured conversation like a Question and Answer session (much like a quiz game) or one-way conversation. When the user must interact with the technology as well as the technology interact with the user – this is where it gets very complex rather quickly. Deep learning technologies may be one way to alleviate some of this – however it's not the complete answer. At least not yet! There will still be some conversational architecting including personality that will need to be integrated into the VUI. Architecting a technology that incorporates VUI takes time and experience – but can be quite rewarding. Now is the time to learn such technologies as they will only become more complex and harder to implement with the advent of ambient computing.

Areas of future work include exploring each of the above technologies deep learning capabilities and gaining more experience in the implementation and architecture of said technologies. Only through additional experimentation and user feedback can this be achieved. Ambient computing is complex and the only way to get better at it is to constantly experiment and get user feedback through direct, methodical research.

References

AWS – Multiple sites:

https://developer.amazon.com/en-US/alexa/alexa-skills-kit/?sc_category=paid&sc_channel=SEM&sc_campaign=SEM-GO^Brand^All^LD^True_Brand^Evergreen^US^English^Text&sc_publisher=GO&sc_content=content&sc_detail=379690597875&sc_funnel=convert&sc_country=US&sc_keyword=%2Balexa%20%2Baws&sc_place=&sc_trackingcode=b&sc_segment=true_phrase&sc_medium=paid%7CSEM%7CSEM-GO^Brand^All^LD^True_Brand^Evergreen^US^English^Text%7CGO%7Ccontent%7C379690597875%7Cconvert%7CUS%7C%2Balexa%20%2Baws%7C%7Cb%7Ctrue_phrase&gclid=Cj0KCQiA0NfvBRCVARIsAO4930kH1SXH2P2tEjJG-6n4YkV1OtpaBdhJQZoLTYLsvyjvBcySVEECEucaAtjyEALw_wcB

https://aws.amazon.com/alexaforbusiness/

https://developer.amazon.com/en-US/alexa

https://docs.aws.amazon.com/lambda/latest/dg/services-alexa.html

https://developer.amazon.com/en-US/alexa/alexa-voice-service

Graphviz – Graph Visualization Software. *http://www.webgraphviz.com/*

Pearl, Cathy., (2017). *Designing Voice User Interfaces.* California: O'Reilly.

Pieraccini, Roberto., (2012). *the voice in the machine: building computers that understand speech.* Massachusetts: The Massachusetts Institute of Technology Press.

Shevat, Amir., (2017). *Designing Bots.* California: O'Reilly.

TWILIO. *https://www.twilio.com/*