SMU Data Science Review

Volume 1 | Number 1 Article 6

2018

Cognitive Virtual Admissions Counselor

Kumar Raja Guvindan Raju
Southern Methodist University, kguvindanraj@mail.smu.edu

Cory Adams
Southern Methodist University, corya@mail.smu.edu

Raghuram Srinivas
Southern Methodist University, rsrinivas@mail.smu.edu

Cognitive Virtual Admissions Counselor

Cory Adams[1], Kumar Raja Guvindan Raju[1], Raghuram Srinivas[1]

[1] Southern Methodist University, SMU Data Science Review, Dallas, TX 75205, USA

Abstract. In this paper, we present a cognitive virtual admissions counselor for the Master of Science in Data Science program at Southern Methodist University. The virtual admissions counselor is a system capable of providing potential students accurate information at the time that they want to know it. After the evaluation of multiple technologies, Amazon's LEX was selected to serve as the core technology for the virtual counselor chatbot. Student surveys were leveraged to collect and generate training data to deploy the natural language capability. The cognitive virtual admissions counselor platform is currently capable of providing an end-to-end conversational dialog to resolve three categories of questions, referred to as intents. These intents allow the chatbot to determine if the user is asking questions about class sizes, tuition amount, and semester start dates. The virtual admissions counselor is able to support three intents, each with accuracy metrics greater than 90%. The virtual admissions counselor is also capable of providing a conversational dialog allowing the successful resolution of potential student inquiries using LEX as the core service.

Keywords: machine learning · Amazon LEX · intent identification · slot variable detection

1 Introduction

In this paper, we present a Cognitive Virtual Admissions Counselor (CVAC) that can receive, understand and respond to questions regarding the Masters in Data Science program offered by SMU. CVAC is a chatbot capable of providing potential student and other interested parties with a natural conversational experience to answer specific questions they may have. A chatbot is a system capable of simulating a human conversation. Normally, they are developed to help provide simple answers or solutions to questions. CVAC provides a better user experience than simply directing user inquiries to a website hosting FAQs. Potential students and other interested parties are able to converse in English with the CVAC chatbot to obtain answers and/or be guided to resources that will help them gain more information and ultimately select the right program.

Millennials represent the majority of students beginning research on prospective universities to determine which institution is the best fit for them. Each individual will likely have different questions, criteria and priorities that will ultimately determine their selection. Universities continue to create and make data available about their institution for these prospective students. As discussed, this data would normally be presented as a generic FAQ article returned from a search engine. Recent technical advancements in machine learning, natural language processing, dialog engines and easily deployable web services has provided the ability to rapidly develop and deploy chatbots. All the major tech companies, such as: IBM, Amazon, and Google, have developed robust web service offerings. Each has a bot framework as part of their offering. Bot frameworks provide off-the-shelf bot basics; intent understanding, slot identification and dialog definition. The user can either use training data provided by the platform or develop its own data, which the framework can easily intake via the user interface or API [11].

Amazon, Google and Microsoft have spent significant resources to offer a "personal assistant" bot as part of their device offering (software on their hardware), Alexa, Google Assistant, and Cortana, respectively. This integrated bot capability has created a world where daily interaction with a bot is normal and the results can be trusted. These bots provide a simple interactive experience of one question and one answer. These interactions or conversations never relied on context for back and forth communications. These simple, yet broad knowledge, "personal assistant" bots led to the progression of human-bot interactions from question and answer, and/or task command completion interfaces to complex chatbots with the ability to mimic a conversation. These "next generation" chatbots are domain specific and require significant training to function. This is a transition from transactional to conversational bots where previous user input is stored and used to demonstrate understanding [10]. This evolution is a function of the ability to easily train and deploy new models. On top of this, bot engines have been developed to track user input, not only to store slot values, but act as a processor. A true conversational bot is able to store user input data and the question context for use later in the conversation. For example: if a user asks "how much does each graduate course cost?" the bot should be able to answer without asking for further information. This demonstrates how an intelligent conversational bot functions vs. a transactional bot. Transactional bots are linear and do not satisfy slots (discussed in section 2.3), while conversational bots understand the slots necessary to resolve an issue. If those slots are filled, there is no need to ask redundant questions [2].

Research shows the younger generation is more likely to use chat applications like WhatsApp, SMS, and Facebook Messenger, to communicate rather than phone calls or other person-to-person direct contact methods. These chat applications make up the majority of communications and are known as asynchronous communication channels. Asynchronous communications allows each party in the conversation to answer at their convenience, where conversation state is stored and available [8]. CVAC was developed based on this fact, people no longer want to communicate in real-time, whether the other side is a human or a bot. While providing CVAC through multiple channels enhances the user experience, it is not the only motivation.

The Cognitive Virtual Admissions Counselor is able to respond to questions regarding admissions, class structure, tuition, and job placement. The CVAC is trained from a database of questions and answers obtained from the Internet, student counselors at SMU, and available FAQs. This research is only meant to demonstrate a capability, solutions provided should not be considered definitive and/or up-to-date solutions. The CVAC is not yet stateful, meaning it has limited ability to cross-reference previous question responses in order to provide the optimal fulfillment for a session. If a potential student decides to enroll or request more information, the CVAC will launch or redirect to the relevant landing page. The best bot offerings available in any domain are those backed by human intelligence.

There is a large potential to reduce costs, increase customer satisfaction and grow potential enrollment rate of students with the use of a Cognitive Virtual Admissions Counselor [4]. A virtual assistant can manage multiple candidates simultaneously, operate twenty-four hours a day, every day of the year. A human operator will only be needed to handle the most complex questions and or questions that the artificial intelligence has not been trained to answer yet. We evaluated multiple bot platforms based on specific criteria and selected a platform to serve as the core technology. We also created domain-specific training data to build the machine learning capability, which we then tested for accuracy.

This paper is organized in the following order, we start with methods and process where we review Natural Language Understanding(NLU), intent identification, slot variable identification, solution flow and resolution paths followed by platform selection criteria and our decisions on platform and 3rd party solution selection. Second part of the paper explains our data gathering and experiment scope, creation and deployment of our chatbot and performance testing. Last but perhaps most important we review ethical considerations of deploying a chatbot.

2 Methods and Process

As stated in the introduction the intent for the CVAC is to provide a platform capable of understanding and resolving potential student's generic questions about SMU academics, financials, and class structure. This research platform will hopefully provide the basis for an enhanced virtual counselor able to handle a wide variety of questions, including pre-requisites, course content, cost, duration, instructors and payment plans for the Data Science course at SMU.

2.1 Natural Language Understanding

A key challenge in creating the CVAC is in classifying the intent of the user's questions expressed in natural language and grouping the type of questions in order to respond accurately. Annotating the corpus of the conversation is subjective and requires judgment [5].As for inter-annotator agreement method, the approach is to baseline the accuracy based on the subject matter the bot is being trained on. One likely method is to conduct cross validation using the entire dataset. This will allow us to run multiple iterations and variations of training and held-out test data to output accuracy metrics based on iteration mean and standard deviation. This could help set a baseline of the expected accuracy of the CVAC.

The CVAC's conversational interfaces will be built using either Amazon LEX, IBM Watson or Microsoft LUIS. All of these provide the functionality and flexibility of natural language understanding (NLU) and automatic speech recognition (ASR). The machine learning (ML) component typically support three types of ML models: binary classification, multiclass classification, and regression [6]. Questions like, "Does SMU offer big data architecture classes?" will be classified using the binary classification model where else questions like "Which data science courses are the most popular among SMU students?" will be classified by the multiclass classification model.

2.2 Intent Identification

Understanding the intent, or the conversation subject, is key to a chatbot as all the following steps and necessary data values will be based on this intent identification. This is why significant time will be spent on training and testing the machine learning models to provide the highest accuracy.

2.3 Slot Variable Identification

Slot values are variables gathered by chatbots to identify the best solution. An intent may or may not have slot variables necessary to invoke the correct resolution, but if provided will increase accuracy. Even if an intent does have slot variables, those variables may or may not be required. Think of slot variables as key-value pairs used after intent identification to invoke the best resolution. A resolution may be URLs, text, emails, content cards, FAQs, etc. leading to an answer satisfying the user input query. Slot variables are specific to an intent and created by the developers based on domain specific knowledge.

2.4 Solution Flow and Resolution Paths

The entire premise for the CVAC platform is to provide correct answers and resolutions to potential students or interested parties. Even if the correct intent is identified and the slot variables filled, the entire system provides no value if it cannot return a satisfactory response to the user.

3 Platform Selection Criteria

The CVAC needs the ability to consume user text and output a user intent to allow the selection of the best resolution response. Several 3rd party services offer machine learning as a service or NLU as part of their bot framework, but there are several factors to consider when selecting which classification service will serve as the core of the bot. Bot solutions are developed to handle a wide range of issues with varying complexity, which results in different service selection based on the use-case requirements. Our CVAC use-case requires a bot classification service capable of receiving user text through a web API and returning results as a response to the request. The request and response should be in a well-defined format such as JSON. Also, since our CVAC will be able to handle a range of questions, the 3rd party service we select must provide a supervised learning option capable of providing either multiclass or multi-label algorithms. Since the underlying learning algorithm will be supervised the service must also allow a simple method for training input (UI or API) and the ability to evaluate the model performance.

The CVAC uses a 3rd party service to train, evaluate, and deploy machine learning models. This service should also allow us to easily create and deploy these models through an endpoint using their service. On top of this requirement, the service should also manage endpoint auto-scaling based on the amount of CVAC classification traffic. This ability allows us to ensure cost minimization by having resources only “spin up” as needed, ensure availability by auto-scaling resources during high customer demand periods, and ensures that time is spent on developing the best possible platform instead of time being spent on the administration of resources.

4 Platform Selection Evaluation

Amazon’s Web Services (AWS) is a widely adopted cloud computing service that recently released a new bot, or conversational interface, service. This service is known as LEX and provides not only a NLU capability, but other bot capabilities allowing LEX to complete conversations end-to-end. LEX has positive and negative aspects to consider as we search for a NLU capability and core platform to drive/manage conversations. Focusing on the NLU capability of LEX, it allows us to define multiple intents and supports significant training with up to 1500 training samples per intent. This allows chat bot developers to easily create and train a model to support multiple intents. However, the documentation does not provide details about the underlying multiclass algorithm and algorithm parameters used by LEX NLU, nor does it output metrics allowing the evaluation of the model performance. They do have a separate machine learning service with three algorithms, one of which is likely the algorithm used by LEX [6]. At this time, a developer is unable to view the precision, recall, and f-score based on training and test data. This is a serious limitation to any bot developer with the knowledge and capability to optimize LEX’s NLU performance. In terms of ease-of-use the service is fully managed and allows simple deployment of the model to an endpoint and since LEX is a fully managed service it scales automatically.

Microsoft offers several cognitive services, but unlike AWS LEX they are decoupled. This allows developers to select the best-of-breed services and integrate each as necessary. Like Amazon, Microsoft offers a bot framework, Microsoft Bot Framework, but unlike Amazon they allow the bot framework to leverage other services for the NLU capability. The CVAC evaluation of Microsoft offerings focused on two services, both having the potential to serve as the core NLU capability. The first is the Language Understanding Intelligent Service (LUIS) which uses a supervised Multiclass Logistic Regression algorithm for intent identification. While the algorithm is known with LUIS, the parameters to this algorithm cannot be changed or optimized by users. However, LUIS does provide the capability to add training data through a REST API and UI. While LUIS does not automatically split training data into train and test for evaluation metrics, it does allow users to split the data and upload a labeled set. LUIS will then output precision, recall, and f-score metrics. LUIS also allows for single click creation of an endpoint and is a fully managed service that auto-scales based on traffic.

The second Microsoft service evaluated was Microsoft Azure Machine Learning (Azure ML) [7]. Unlike LEX and LUIS, Azure ML is a very powerful NLU platform designed for advanced users. Within Azure ML a user can upload training data from multiple sources, transform data, implement feature selection, select from 5 multiclass machine learning algorithms, tune parameters for each algorithm, run custom Python and R code inline for data manipulation, plotting, etc., and leverage built-in text analytics capabilities. Unlike LUIS, this service is not designed for simple bot development use cases or developers new to machine learning. Azure ML is a powerful service that provides the capability to evaluate and compare developed models. Every aspect of the model development can be controlled and the easily exposed through the simple web service deployment options. Azure ML will also scale automatically based on traffic.

IBM's Watson offering includes a Natural Language Understanding and a Natural Language Classifier service. The Natural Language Understanding service provides the ability to categorize text input into categories, along with the ability to analyze several other semantic features (entities, keywords, sentiment, etc.) [9]. The documentation shows this service uses a five-level classification hierarchy, but the specific categories are predefined. Since this CVAC will need flexibility in the number and depth of the classification output, this may not be the best service for our specific research project. The Natural Language Classifier service does offer the ability to create and train classification models based on custom labeled training data. There is a 15,000 limit to the number of training samples allowed, which is more than enough for this research project. Like the other managed services, IBM's Natural Language Classifier service is exposed through a well-defined and documented API [9].

5 Selection of 3rd Party Solution

Amazon's LEX service was selected as the NLU service for this research project. While LEX does have a limitation in the number of training samples allowed per intent (1,500), this limitation does not impact our research given the number of training samples currently available (discussed further in Training Data Generation and Intent Selection section below). LEX provides all the necessary requirements for this research:

- NLU capability for Intent identification and Entity extraction
- Fully-managed service capable of scaling automatically
- High availability
- Ability to drive/manage conversations
- Web API request and response
- Well-defined format and documentation for communications
- Session handling (slot storage, intent attribute share, session timeout)
- Integration with 3rd party messaging services (ex. Facebook, Slack)

LEX is an easy-to-use complete bot platform service enabling developers the ability to leverage the robust capabilities provided by the AWS platform and its significant service offerings. The simplicity of the LEX service was a driving factor in the selection as LEX provides not only the NLU capability and requirements listed above, but also allows for the development of conversational dialog within the same service. Several of the other cloud platforms evaluated separate their NLU offering from the bot platform offering. Separating these functions does allow users to integrate best of breed services, but it also increases the barrier of entry for skills necessary to build and maintain an end-to-end bot solution. Since the goal of this research is to provide an end-to-end solution to be easily managed and improved, minimizing administration requirements while providing a powerful capability was a key factor in the selection process.

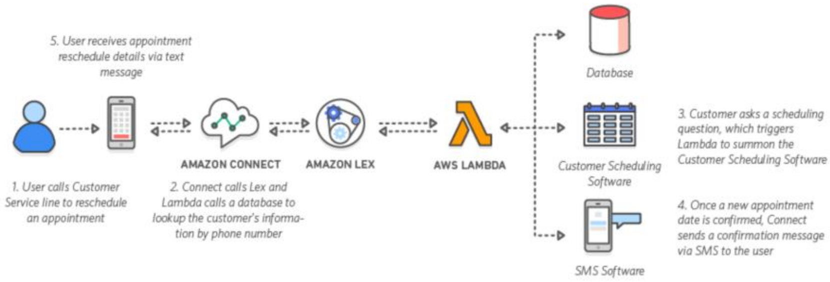

Figure 1: Amazon LEX architecture [12]

6 Data Gathering and Experiment Scope

6.1 Data Collection

This platform will enable prospective students to ask questions and receive answers, providing the necessary information they require in their planning and selection of a university and program. While CVAC will not initially be capable of resolving all student inquiries, it will be architected to scale and continuously learn, with the goal of providing increased capabilities over time resulting in greater value and support. For the first iteration of the platform the types of questions CVAC will be able to resolve, we will refer to these questions as intents, will be determined based on likelihood of intent volume. In other words, the initial platform will answer questions that are likely to have the highest volume. To provide the greatest level of value our focus will be on identifying and selecting common user intents. In addition to these inquiries being frequent, they should also only be proposed for selection only if the intent can be resolved through CVAC dialog. Complex, ambiguous, and questions requiring PII data will not be supported by CVAC at this time.

Questions were generated by asking a group of graduate student at SMU to provide the top 25 questions they had prior to their selection of a university and program. The survey results were then used to group common questions into three generic categories. These questions were then manually classified into the selected three categories. These categories are broad and can encompass multiple intents. For this experiment researchers used these easily differentiated intents to provide consistency during training and optimize classification accuracy.

The basic capability all chatbots need is the ability to accurately identify the user intent. Once the intent is identified, the system can be programmed to ask follow-up questions to fill slot variable values in order to navigate and return the best flow resolution. Next we will further define which intents we selected for this research, as well as further information on what slots are and how they can be used to provide the optimal resolution through conversational dialog.

6.2 Manual Intent Definition

The top three categories and potential intents were identified from the survey result manual labeling process as:

1. General Academics – time commitments, class sizes, semester length and semester start dates, job placement

2. Financial Information – costs per course, tuition assistance, VA benefits and payment methods

3. Admissions – application deadlines, how to apply, prerequisites

For the scope of this research we decided to focus on one intent per category. For the general academics category we will create the intent "Class_Structure." For the financial information category we will create the intent "Tuition." For the admissions category we will create the intent "Admissions_Requirements." This will allow us to focus on how well the system performs across differentiated topics and spend more time building out the slot identification and conversational dialog capabilities.

6.3 Slot Creation

LEX offers built-in slot types CVAC can leverage without needing to build our own. This is useful, especially when building a product around a common domain and can simply select prebuilt slots. CVAC was not able to leverage many of the built-in slots, but can leverage convenient slots variables like yes and no parsing. CVAC specific slot variables, those were created by providing the slot a name and a list of expected values. All slot variables used in CVAC are required to be filled before it is capable of returning a response. Thus, we limited the number questions to only those necessary to capture required fields. Our belief is that by making the conversation as efficient as possible we enhance the user experience.

Table 1. Example breakdown of CVAC supported intents and assigned slot variables.

Intent	Slot Variable	Sample Expected Values
Admission Requirements	Semester	Fall, Spring, Summer
Class Structure	Class Type	Online, On Campus

Tuition	Payment Options	Financial Aid, VA

6.4 Conversational Dialog

CVAC is a conversational bot. The goal of conversational bots is to identify the user intent and identify slots to provide a solution or resolution. Web forms may be able to accomplish the same goal, though it will be less personal.

To differentiate between web forms and conversational bots, users must be able to input text at any point in the conversation, much like people in a conversation. By allowing arbitrary input throughout the dialog we must ensure the LEX NLU algorithm is capable of understanding and correctly identifying the intent and/or slot variables. By assigning and understanding the intent and slot values we are able to track the dialog state and only return specific questions at specific points in a flow.

LEX provides the framework for a mostly linear conversation. First, what is the user question and returned intent? Next, are there any slot values? If so, ask the questions and fill the slots based on the user response. Finally, return a response associated with the intent. This works well if users clearly answer questions and do so in the order they are received.

Issues arise when users are vague, do not answer questions with an expected response, want to change their answer, or want to change the intent itself during the conversation. This is a very difficult task to solve and the focus of current research around dialog state tracking. The ability to use previous information to provide context to current input allows, the system to choose the best response based on the available data [1]. While LEX does not provide specific support for this type of capability, limited functionality was built into CVAC using AWS Lambda functions, hooked after each user input. This allowed us the ability to create custom logic outside of LEX.

7 Creation and Deployment

Amazon provides the ability to build, configure, and deploy LEX through the AWS dashboard. Amazon also provides these same abilities to be completed using the LEX API. Since CVAC is a single bot and a limited number of intents, the dashboard was used to create the bot and define the intents. All bot related functions associated with bot actions and data type creation/updating can be achieved through the dashboard. While our bot is research focused and the needs are not the same as an enterprise chatbot, it is important to understand LEX limitations and deployment best practices. The AWS dashboard provides a simple-to-use user interface, but does not provide a scalable solution for an enterprise chatbot. LEX developed bots may have 100 intents, each of those intents may have up to 1,500 utterances, and the bot has a maximum of 50,000 slot type values. This research bot did not approach these limitations, but in many cases using the APIs provided by LEX improved the development process through allowing management of the bot through scripts to quickly update the bot capability. The chatbot was setup to only understand domain specific questions and uses LEX as a service for intent identification, slot variable selection, and solution flow. As discussed earlier in the Selection of 3rd party solution section, you cannot mix and match these technologies when using LEX as the chatbot core. Lex does not separate out these capabilities.

CVAC training data consisted three intents with fifty sample utterances per intent. As the three intents do not appear to have overlapping samples or a need to provide a large number of samples to differentiate between intents, we believe this number of samples will be sufficient to achieve high accuracy metrics for intent identification. Using the Python "boto3" AWS library we were able to connect to the client and upload the training data. The next section will discuss why this ability to quickly update the training data associated with an intent is important in performance evaluation.

8 Performance Testing

In the evaluation section we pointed out the fact LEX is not capable of providing an accuracy metric based on the training data provided to it. In order to be able to identify how well a bot is performing, we measure how it classifies intents. Knowing intent accuracy metrics will allow us to focus on weak or underperforming intent classes. Thus we may be able to add more training samples if there is a large imbalance or other issues. Since LEX does not have this capability built-in we had to build our own testing code.

To evaluate CVAC, we use the standard practice of evaluating the accuracy using cross-validation. Cross-validation is the process of configuring folds (runs) and splitting the data set into training and testing datasets based on the number of folds. Each fold uses a different seed to create different testing and training sets containing a new combination based on the same overall dataset. Accuracy is a generic term here, the true performance metrics we output and monitor to evaluate the intent classifier performance are precision and recall.

$$\text{Precision (P)} = \frac{\text{tp}}{\text{tp+fp}} \; . \tag{1}$$

$$\text{Recall (R)} = \frac{\text{tp}}{\text{tp+fn}} \; . \tag{2}$$

In the precision and recall equations "tp" is the count of true positives returned from the training set prediction, per intent. This means these utterances were labeled correctly by the intent classifier. Precision uses "fp" which is the count of false positives returned from the training set prediction, per intent. For each intent, these utterances were incorrectly labeled as this intent. Recall uses "fn" which is the count of false negatives returned from the training set prediction, per intent. For each utterance that was incorrectly labeled, the correct intent would then be a false negative.

Table 2. Intent classifier testing results using 10-fold cross validation.

Fold Number	P(Admissions)	R(Admissions)	P(Class)	R(Class)	P(Tuition)	R(Tuition)
0	1	1	1	1	1	1
1	1	1	1	1	1	1
2	0.875	1	1	0.8571	1	1
3	0.75	1	1	0.875	1	1
4	1	1	1	1	1	1
5	0.8	0.8	1	0.5	0.9091	1
6	1	1	0.8571	1	1	0.5
7	1	1	1	1	1	1
8	1	0.8333	0.875	1	1	1
9	1	1	1	1	1	1
Mean	0.9425	0.9633	0.9732	0.9232	0.9909	0.95

While the results shown in the table above are promising, additional training will be added to improve the accuracy. This training/testing dataset was gathered by current students, but once deployed the training data will increase with the goal of a continuous learning platform. As prospective students use CVAC the NLU understanding will improve with the addition of new questions. New questions can be labeled and added to the training data, resulting in increased accuracy of currently supported intents, while allowing analysis of frequently asked question to nominate potential intents. Another piece of data we would like to be able to access is the features and the feature weights. By knowing these values we could understand why misclassifications are happening and additional training as necessary. Another metric not discussed in detail is latency. This was not evaluated in-depth as CVAC deployed using LEX responds almost immediately. Further research may include whether users want an instant response or expect a delay, as they expect during a traditional conversation.

9 Ethical Considerations

This paper has discussed how and why we developed CVAC and the motivation for doing so. This paper would not be complete without addressing ethical considerations associated with the creation and deployment of CVAC. Ethical questions surround the AI industry as a whole, but each AI based platform should only address ethical issues within the context of the technology, data collection, and audience. CVAC does not approach the technical, nor the data collection capabilities of AI products offered by large companies like Google or Apple. The following questions and answers summarize the ethics concerns raised during the research project along with the team's responses.

How does CVAC impact SMU staff, will this technology replace humans?

Call centers normally employ the smallest number of agents, with the goal of answering customer questions, but focusing on efficiency. Efficiency in this case is the amount of time an agent needs to satisfy a customer query. CVAC is meant to serve as a resource capable of answering common questions, while allowing human agents the ability to spend more time on complex issues. If customer service agents do not feel the need to rush a session, they will be able to provide a better user experience for potential students.

Does CVAC gather personally identifiable information (PII) data?

There will be minimal requirements to store personal data of students, potential students or their parent guardians. The only data that will be stored by the CVAC will be names of one of the parties described before, contact information of their choosing, either email, phone, address, age and if their parents or potential students. This information will only be requested after interactions with the CVAC. This will remove any emotional barriers that candidates may have in giving out their personal information prior to getting the information they needed. There will be no financial, grade, religious or medical information requested or stored. All session information except that required to contact the potential student will be deleted or anonymized. Anonymous data will only be stored for the purposes of analyzing the effectiveness of the session and understanding the key performance indexes of the CVAC such as engagement rates, confusion triggers, conversation steps, average number of conversations per user and, and only for a 30 day period. Any contact information stored will not be shared or sold to parties outside of SMU.

Who is responsible for the content accuracy provided by the virtual counselor?

CVAC is expected to enhance and supplement SMU staff members whose job it is to respond to these queries allowing for basic information to be available anytime and all the time. This tool will allow SMU staff to focus on queries of a more difficult and complex nature. CVAC is a research proof-of-concept developed by this team during the MSDS program to satisfy an academic requirement. Any future development and/or production deployment belongs to the SMU program. CVAC is meant to demonstrate how and why a chatbot is beneficial in the academic domain, while including the limitations and issues.

10 Conclusion

In this paper, we presented the CVAC service we built in part to research the current state of the chatbot technology and deliver a new communications channel for SMU to interact with potential students. The motivation for this research is the heavy adoption of asynchronous channels for human-to-human conversations, computer-to-human dialog research is clearly an important next step. This paper outlined current technology and major bot frameworks available in the market today. We defined evaluation criteria and spent significant time comparing IBM, Amazon, and Microsoft services to select the best platform for our use case. Amazon's LEX service was chosen as it met the evaluation criteria and enabled us to build and deploy quickly with an easy to navigate UI and well-define APIs.

CVAC was developed to provide end-to-end resolution, from intent identification to query fulfillment. The intent identification model used supervised machine learning based on training data created by SMU graduate students. Three intents were chosen to demonstrate the process and potential capability CVAC offers. Each of these intents were trained and test using cross-validation, all providing accuracy metrics greater than 90%. CVAC should be viewed as a successful research project providing a researched and implemented technology to build upon.

References

1. Yu, K., Chen, L., Sun, K. et al. Evolvable dialogue state tracking for statistical dialogue management. Front. Comput. Sci. vol 10, issue 2, pp 201-215 (2016)
2. Radziwill, N., Benton, M.: Evaluating Quality of Chatbots and Intelligent Conversational Agents. arXiv:1704.04579 [cs.CY], June 2017
3. IBM developer works – Build a hybrid cloud solution with IBM PureApplication and Bluemix, https://www.ibm.com/developerworks/library/mw-1608-grant2-bluemix-trs/
4. TALANIA white paper, July2016 (https://chatbotsmagazine.com/why-chatbots-work-for-businesses-6dc7a050c9c0)
5. Inter-Coder Agreement for Computational Linguistics (Ron Artstein∗ University of Essex Massimo Poesio∗∗ University of Essex/Universita di Trento)
6. Amazon Machine Learning – Types of Models, http://docs.aws.amazon.com/machine-learning/latest/dg/types-of-ml-models.html
7. Microsoft AI Platform – Build Intelligent Software, https://azure.microsoft.com/mediahandler/files/resourcefiles/5467086f-7c5a-4106-8615-7f5e4ad217fd/microsoft-ai-platform-whitepaper.pdf
8. Eeuwen, M.: Mobile conversational commerce: messenger chatbots as the next interface between businesses and consumers. Journal of College Admission, n195 p20-25 Spr (2007)
9. IBM Cloud Docs- Natural Language Understanding, https://www.ibm.com/watson/developercloud/doc/natural-language-understanding/#categories
10. Bhargava, A., Celikyilmaz, A., Hakkani-Tur, D., Sarikaya, R.: Easy Contextual Intent Prediction and Slot Detection. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 8337-8341, IEEE Press, New York (2013)
11. Amazon Lex Developer Guide, http://docs.aws.amazon.com/lex/latest/dg/lex-dg.pdf
12. Amazon Lex conversational interfaces for your applications https://aws.amazon.com/lex/