ACEbot – A University Chatbot

Rajamalli Keerthana R[1], Fathima G[2]

[1]P.G. Scholar, Department of Computer Science and Engineering, Adhiyamaan College of Engineering, Hosur, India.
[2]Professor, Department of Computer Science and Engineering, Adhiyamaan College of Engineering, Hosur, India

***

Abstract- *Chatbots are conversational software that helps in conducting a conversation via textual or auditory methods with customers. Chatbots are now widely used in almost all customer service stations and for information acquisition. Chatbots provides a way to answer all customer questions without exploiting human resources for basic inquiries. Chatbots have been under research for improving its response rate and to make the AI assistant seem more human. Over the past years, there have been various Chatbot models and methods proposed all of which strive to increase the response rate and give character to the chatbot. Universities across the globe have incorporated this technology in order to better their student-management interactions. The university chatbot mainly focuses on providing the students, information which are not easily available thus enhancing the knowledge of the students on university-related matters. The university chatbot also helps in resolving various queries of new students, who might find it difficult to search for certain answers. The university chatbot also helps in reducing the tasks at the information desk. The almost-human interaction in the chatbot makes it easy for the students to feel like they have been conversing with an information desk personal. The university chatbot can provide common information like the fee structure, faculty detail, curriculum detail, etc., and more rarely found details like contact information of faculties, donation queries, academic performance related queries, etc. This paper explains in detail the working mechanics of the chatbot created for the University and gives a detailed explanation of various chatbot models used.*

Keywords- Artificial Intelligence, Chatbots, University chatbots, NLP, Use-cases, SEQ2SEQ.

1. INTRODUCTION

Artificial intelligence or machine intelligence is a display of decision-making capabilities by machines much similar to humans. Artificial intelligence is a field where a new phenomenon is discovered every day. Intelligent systems have proven to be the best innovation by mankind in this generation. There has been significant relief in various commercial fields which have incorporated Artificial Intelligence into their technology or work process. Artificial Intelligence systems has the ability to absorb information from the environment and learn from them in order to make decision in various situations.

Chatbot is a conversational software that employs both Artificial Intelligence (AI) and Natural Language Processing (NLP) to communicate with humans in a human understandable language. NLP and AI have one important task among its various other tasks to process data and that is, Modelling conversation. Since AI was found, there has been incredible changes in the IT field, but this also means there has been various challenges in the field of AI. One such challenge is creating a strong, reliable and Knowledgeable chatbot. Chatbots have been integrated with almost every field as they understand a user's query and provide the solution for the problem. Modelling conversations remains one of the toughest challenges in AI. In the past, methods for creating chatbot structures relied on handwritten rules and templates or simple statistical methods. With the development of deep learning, these models were quickly replaced in 2015 by neural networks that could be trained from end to end. While they may not be perfect Google's numerous applications use chatbots along with other tech giants, For example - Google Assistant [Google, 2017], Apple's Siri [Apple, 2017], or Microsoft's Cortana [Microsoft, 2017a].

There are four main components of chatbots - Natural language processing, Dialog manager, Content, Custom Integrations. The first component is Natural Language Processing. It is responsible for analysing user requests, i.e. it basically takes an unconfigured phrase sent by the user and converts it into structured data, which is then handled by the chatbot codebase. Such a component can be built in-house, but using a third-party service is much more interesting. Natural language processing is a task that does not change from one chatbot to another. DialogFlow and Wit.ai are commonly used tools for NLP as they are easy to use and free. The second component is the Dialog Manager. This component is responsible for determining what to say to the user based on the user input, user's past interactions and other data that can be pulled from other sources. The Dialog Manager can be a series of IF statements or something more complex. Similar to the natural language processing component, the dialog manager component can also be created in-house or using an off-the-shelf solution. Dialogflow also helps with the dialog manager component. The third component is content, which describes the template for what the chatbot application is going to say once it has decided what to say. This component cannot be created using an off-the-shelf solution as the chatbot will change completely depending on the nature and requirements of the chatbot, for example, a university chatbot content is not the same as a customer service chatbot content. The content structured by this module will affect how the user views the chatbot from the user experience and marketing point of view. This component can be created locally or with the use of third-party tools because it allows the content to be redone and previewed before creating the chatbot. In this way, we can ensure that communication is structured in a way that provides the best possible user experience based on the business objective. Botsociety is the best third-party tool for creating this component. The fourth component is the Custom Integrations. This component is not absolutely required in the chatbot. Usually, more powerful and meaningful bots use these elements. This component pulls data from the database of the web service or chatbot database and then notifies the dialog manager of the conditions to check if the request has been made correctly. The 'webhook' feature of the dialog manager is used to create a dedicated port for implementing these components.

All these components work together to create a chatbot that can converse with a user imitating a human-human interaction.

2. CHATBOT MODELS

Chatbot models typically take normal language phrases uttered by a user as input and responds with an answering phrase as an output. There are two main approaches to generating answers or responses. Hardcoded templates and rules are the traditional approaches that are used to create chatbots. The most innovative approach is the application of deep learning. Neural network models are trained with large amounts of data in order to learn the process of generating appropriate and grammatically correct answers for the user input. Models are also tailored to suit spoken or visual inputs.

2.1 NEURAL NETWORKS

In Machine Learning, there is a key concept that differentiates Rule-based learning approach from Network-based learning approach, that is, the existence of a learning mechanism in neural network-based approach. Deep Learning is a subfield of Machine Learning and thus it has to be distinguished clearly from traditional Machine Learning. Neural Networks are the backbone of conversational modelling. Each word (symbol) must be converted to a numerical representation when using neural networks for natural language processing (NLP) tasks. This is done through word embedding, which refers to each word as a constant quantity vector of real numbers. Embedding of words is useful because instead of treating words as large vectors of vocabulary size, they are represented in very small dimensions. Instead of using handwritten rules, deep learning models convert input sentences directly into answers using matrix multipliers and non-linear functions with millions of parameters. Neural network-based conversational models can be divided into two categories such as retrieval-based models and generative models. The retrieval-based model gives an answer from the database by calculating the cosine similarity between the embedded words and the candidate responses or by calculating the answer to the current input sentence based on a scoring function, which can be implemented as a neural network. The other type of conversational model, namely, the generation model, responds to one work at a time as it calculates the probabilities in the entire vocabulary. There are also approaches that integrate the two types of dialogue systems by comparing the generated response with the retrieved response and determining which is the best replacement.

2.2 RECURRENT NEURAL NETWORKS

A Recurrent neural network (RNN) is a class of artificial neural networks where the connections between nodes form a temporal line-driven map or a directed graph as depicted by figure 1. This allows for the expression of temporary dynamic behaviour. Recurrent Neural Networks are derived from Feed Forward Neural Networks; therefore, they use the internal memory to process inputs of variable lengths. This type of RNN tasks can be applied to handwriting recognition or speech recognition. A directed acyclic map which can be changed into an unrestricted feed forward neural network is called a finite impulse network while an infinite impulse recurrent network is an unrolled directed cyclic graph. Finite impulse and infinite impulse recurrent networks can have additional storage levels, and the storage is directly controlled by the neural network. Both infinite impulse network and finite impulse network can be referred as RNN as they both exhibit transient dynamic behaviour. If time delays are included or have feedback loops, the storage can be replaced by another network or map. Such restricted states are referred to as gated state or gated memory and are part of long short-term memory networks (LSDMs) and gated recurring units. It is also known as the feedback neural network (FNN). Usually long short-term memory (LSTM) or gated recurrent units (GRU) are used for the activation function. LSDMs were developed to combat the problem of long-term dependencies faced by vanilla RN. As the number of unregistered steps increases, it becomes difficult for simple RNs to learn to remember information found many steps earlier.

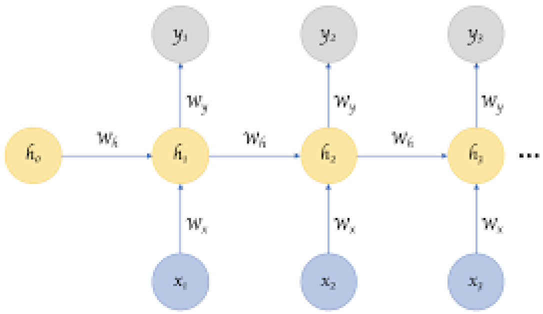


Fig. 1. Recurrent Neural Network

2.3 SEQ2SEQ MODEL

The Seq2seq model revolutionized the process of translation through the use of deep learning. It not only takes into account the current word / input when translating, but also the word / input adjacent to it. Seq2seq model uses a Recurrent Neural Network to take a sequence of input words or sentences and then creates a sequence of output words or sentences. The RNN used is the advanced version, that is, LSTM or GRU is used. It creates the context of the word by taking two inputs at each stage. One from the user and the other from its previous output, hence the recurring name (output goes into the input).

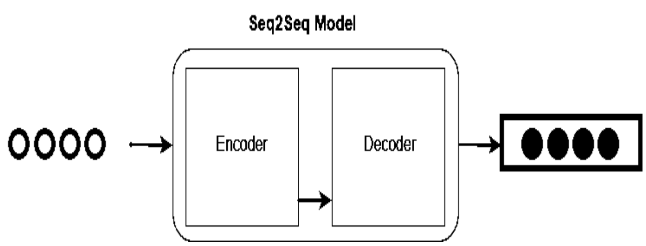


Fig. 2. Seq2seq model

Figure 2 depicts a Seq2seq model. Seq2seq model is also called as an Encoder-Decoder model because it has two main components, the Encoder and Decoder. The encoder uses deep neural network layers and converts the input words into corresponding hidden vectors. Each vector represents the current word and the context of the word. The decoder is similar to the encoder. It takes the hidden vector generated by the encoder, its own hidden positions and the current word to produce the next hidden vector as input and then predicts the next word.

3. LITERATURE SURVEY

In paper [1], an open-domain response generator has been developed with personality and identity that mimics characters from popular TV shows: Barney from How I Met Your Mother, Sheldon from The Big Bang Theory, and Michael from The Office and Joey from Friends. There are numerous uses for this model, such as allowing individuals to talk to their favourite celebrities, creating AI assistants that are more life-like or creating virtual alternative-egos about ourselves. Model [1] was trained from end to end without any hand-crafted rules. The Bots created are reasonably fluent, have unique personalities, and seem to have learned some aspects of their identity. The results of standard automated translation model evaluations of model [1] gave very low scores. However, after evaluating the model [1] with a person's judgment component, chatbot worked better to make more than 50% of the users believe that a man responds in the bot, and that the answer came from a real human (or the tv personality used). As proposed in [1], it is important to add personality to chat bots to keep the user comfortable.

Matching compatibility with its multi-turn environment can be a major challenge in recovery-based chats. Current studies create multiple representations of context and response to facilitate response selection, but they use these representations in isolation and ignore the relationships between representations. To address these issues, Paper [2] proposes a hierarchical aggregation network of multi representation (HAMR) to adequately enhance numerous representations and enhance valuable information. The model proposed in [2] uses bidirectional recurrent neural networks (BiRNN) to extract syntactical and linguistic representations of sentences and uses a self-coordination mechanism to mix these representations. Model [2] fuses completely different matching information between every utterance in context and response, using an identical aggregation mechanism that is produced by an attention mechanism.

The authors of paper [3] propose methods for learning symbolic processing with deep learning and developing question-answer methods through learned models. Learning through symbolic processing, deep learning models, neural machine translation (NMT) and Word2Vec training performed by Prolog processing systems that run integration, resolution and list functions. Implementing a Prolog-like process using deep learning is a new experiment that has not been conducted before paper [3]. The results of paper [3] experiments revealed that the methods proposed in model [3] as a result of the symbolic process are superior to the traditional ones, which have rich representations, can explain the inputs even if they contain unknown symbols, and can be learned with a small amount of training data.

In the paper [4], the answer acquisition method for Knowledge Base Question Answering (KBQA) systems based on the Dynamic Memory Network is proposed, in which representative learning is used to represent the natural language questions and knowledge base subsections raised by users of related entities. These representations are taken as the inputs of the dynamic memory network. Correct answers are obtained by using memory and inference skills. The test results demonstrate the effectiveness of the approach taken by the authors of paper [4]. Table 1 discusses the results of each model described in [1], [2], [3], [4] and their training datasets.

Conversation dialogue response development is a challenge in chatbot applications. Recently neural-network based conversational models, including seq2seq model and RNN language models, have been able to generate fluent and grammatically compatible responses, while most of the responses generated by these models have a large range of chit-chat style instead of being informative. After examining the models currently in use, the paper [5] authors found a primary challenge of modelling and generating communication terms such as named companies, especially when there are few companies on the training corpus. To solve this problem, [5] proposes to enhance the neural network-based production structure with knowledge embedding and knowledge-focused reader to incorporate external text knowledge into the conversation model to facilitate conversation modelling and generation.

The automated chatbot has recently attracted a lot of attention and can be used on many applications such as personal assistants, customer service for online shopping or technical support on a smart phone. One of the most fundamental challenges in implementing a successful chatbot system is how to effectively generate the correct answers given for the input question. Lots of earlier chatbot models focused on rule-based or event-based mechanism. For example, Banchs and Li [6] introduced an event-based dialogue system based on a large dialog database that uses a dual search strategy to complete a conversation. Similarly, Wang et al. [7] Proposed a recovery-based automated question-and-answer model by creating a short text dialog set. Although the recovery-based approach has achieved

satisfactory performance and is easy to implement, it has a limit when the input question does not fall within the scope of
the existing question and answer repository. For the development of deep learning technology, especially for the emergence of end-to-end learning methods, a lot of advanced response generation based conversational systems have been proposed. The SEQ2SEQ model is one of the most popular approaches and has been widely praised as a promising solution for this task [8]. A typical SEQ2SEQ model is an encoder-decoder architecture consisting of recurrent neural networks (RNNs) that encode the source report and map it to a fixed length, while the vector space is decoded by another RNN to the vector location of the target statement [9]. Although the SEQ2SEQ model is widely used and can produce smooth, grammatically consistent responses [10], [11] it has a significant limitation, or so-called "secure answer" problem [12], which makes model [13] more responsive (a. E.g., "I do not know", "Yes, that's right"), thus making such chatbot [13] far from practical use.

MODEL	**[1]**	**[2]**	**[3]**	**[4]**
Working	Basic chatbot with personality	Choose appropriate response based on multiple context	Question answering system that implements deep leaning	Question answering system that uses dynamic memory
Technology used	Seq2seq model	Bidirectional recurrent Neural Networks	Neural Machine translation	Dynamic memory network
Feature added	No end token in seq2seq model	Hierarchical aggregation network of multi representation (HAMR)	Prolog-like process with deep learning	Dynamic memory network
Drawback	Fails at generating longer responses	Number of turns in context affects the performance	-	-
Data set used	Cornell Movie-Dialogs Corpus and dialogs of characters from TV shows	Ubuntu Dialogue Corpus and Douban Conversation Corpus	Geoquery dataset	Kinsouces and geoquery

Training data	280,000 pairs	1 million context-response pairs	1256 data pairs	14,609 question answer pairs
Test data	20,000 data pairs	50,000 pairs	Yes/no questions	9,870 question answer pairs
Accuracy	50%	88%	82.3%	94.41%

Table 1
Comparison of different models

4. PROPOSED MODEL

Every year the students are excited to choose a university or college that will help them achieve their dreams and gives them a variety of benefits. They are determined to know every tiny detail of the colleges they are interested in before applying. Thus, for any college or university it is important to communicate promptly in order to turn interested applicants into registered students. The most asked questions from the students are the courses available, selection criteria, fees, accommodation and transport facilities. This is a busy time for the admission departments, and it takes a lot of time and resources to answer these questions quickly.

All the enquires about the university are not immediately or adequately answered due to short staff. This is where university chatbots comes into play. The proposed system is a university chatbot that can answer all the queries of a student. The proposed system is created to assist the university in the admission process and information sharing. The proposed model, ACEbot, uses Natural Language processing (NLP), and applies it to Seq2Seq model. Each word from the student is taken into account and needed information is extracted from the user input. The extracted information is then used to find a match and then reply to the query with appropriate response. The direction of the flow of conversation is represented using a flowchart, Fig.2. The implementation details of the ACEbot are discussed in detail in the following section.

5. IMPLEMENTATION

5.1 SOFTWARE

The site used to create the chatbot is SnatchBot. SnatchBot is an enterprise-grade, bot-building platform service that was developed with a central principle in mind: to make the bot-building accessible to anyone whether it is a developer, enterprise or individual with no coding skills required. SnatchBot works with small start-ups to companies that are leading the tech world. The SnatchBot clients use their site to streamline business workflows and communications with a single message-based interface that does not require coding skills. With their omni-channel platform, SnatchBot's tools support the entire life cycle of a chatbot, from development and testing to deployment, release, hosting, and monitoring. They include Natural Language Processing, Machine Learning and voice recognition. The site provides sophisticated management features with organizational-quality security in compliance with all regulatory directives.

5.2 FLOWCHART

The chatbot build in SnatchBot gives the flow of the chatbot. New interactions can be added to the bot where the message from the bot can be specified along with the option of text-to-speech conversion. The welcome message of the chatbot sets the tone of the conversation. Therefore, the welcome message is followed by an inquiry of name. Adding some conversational dialogs to the chatbot helps in increasing the character of the bot. This gives the bot an almost-human feel. Every student has some common queries that are asked all the time. These common queries can be put in use cases in order to build a foundation for the bot. The queries are linked according to the answers expected. These connections help in keeping the conversation flowing. The flow chart used for ACEbot is defined in figure 3.

The first message to be displayed is the welcome message. The welcome message consists of a formal greeting. Following the welcome message, some conversational dialog must be present like asking for the student name. Once the pleasantries are done, the Bot provides options for the student in order to reduce the burden of typing for the student and also to provide perspective for the student. The query options can include Fee structure, Faculty detail, result, Other. These options answers variety of common questions asked by students on the daily. If the student chooses fee structure then the student is asked to specify their degree and year, if results is chosen, the student register number is required, if other is chosen then the student can ask their own question based on which the next flow is created. Based on the answers the Chatbot provides the required information.

The required information is extracted from the answers of the students using the variables and attributes. The required variables are set, and the attribute is defined. This helps in identifying the information needed from the answers. The chatbot is then trained to identify different sets of answers that contain the same query using NLP.

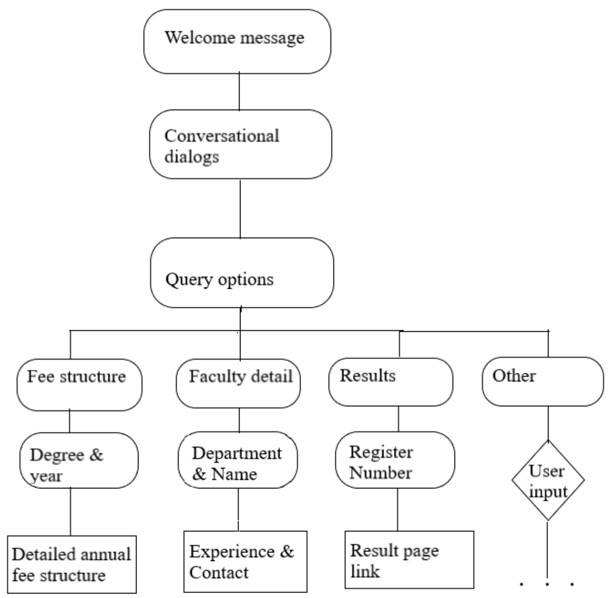


Fig. 3. Conversation Flowchart

5.3 NLP

Sophisticated, natural language processing capabilities are used to make the chatbot do more than just follow the branches of scripted conversation. An NLP is a tool that allows a chatbot to understand the meaning of a user's statement and act accordingly. Essential part of the sentence has to be considered when a chatbot has to break down the meaning of a sentence. An effective way to do this for the broader community of researchers in artificial intelligence is to differentiate between organizations and purpose. An entity in a sentence is an object named in the real world, people, places, systems, times, and so on. For example, in the sentence: My sister went on vacation to New York in 2017, the entities are sister (person), New York (location) and 2017 (time). The NLP models in SnatchBot are great at identifying entities, and they can do that with human accuracy. Intent in a sentence is the purpose or objective of the statement. In a sentence of the same genre, whether I want to buy a one-year membership or book an appointment, the intent is easy to identify, i.e. buy and book, respectively. However, many sentences do not have a clear intent, therefore, recognizing the intent of a chatbot is very challenging.

Since the NLP model is developed from scratch it will be very basic at first. Several examples of sentences must be provided manually, with information about what entities are in the sentence or what the purpose is. Obviously, the NLP model should draw many more examples so that it can perform better and be very accurate. Users need more models to deliver accurate results, but if configured correctly, the intents will act like a charm. In order to build the NLP model over time, the chatbot has to learn, especially from its flaws, so that it becomes more and more precise in solving the task to be observed. Machine learning is an exciting and widely discussed topic in the search for true artificial intelligence. SnatchBot models refer to machine learning in the sense that based on the example sentences provided and their effects, the model will make decisions about the new sentences it will encounter. The platform also offers what is sometimes called supervised machine learning. In light of the data from various conversations, it can be concluded that the chatbot requires more training, including the complex sentences that have been identified, and make the right decision to bode with when examining the sentence. Supervised machine learning will provide a higher success rate for the next round of supervised machine learning. Cycling between supervision and doing the evaluation of sentences independently will ultimately result in a more refined and successful model.

5.4 DATASET

The database used here is a collection of WhatsApp conversations collected from the University Admissions Administrator, Phone conversations from admission officers along with data collected from the University website. The number of conversations obtained is 4,953. It is then divided into a training set of 3516 conversations and a test set of 1437 conversations. Each conversation has one or more questions as input sentences and one or more answers as target sentences.

5.5 BASELINES

The proposed work is compared to other models discussed in section III of the paper (Literature survey).
[A] A Neural Chatbot with Personality

To test this model, the authors use both automated measurements and human judgment. BLEU and ROUGE measurements are used for automatic measurements. These are popular measurements commonly used for translation models. The BLEU metric uses a modified n-gram accuracy score that attempts to model human judgment on how accurate a translation is. The ROUGE-S metric uses the interplay of a skip-bigrams between a candidate response and a reference answer, which measures how similar the sample response is to the expected script response.

[B] Multi-Turn Response Selection for Chatbots With Hierarchical Aggregation Network of Multi-Representation
In this research, the authors use Rn @ k as the main evaluative measure. Rn @ k refers to the recall of a positive answer among the best k selected candidates, and the positive answer means that the best k is ranked top among the n candidate answers given. This model measures the performance of the Ubuntu dialog corpus using R10 @ 1, R10 2, R10 @ 5 and R2 @ 1. Apart from Rn @ k, average accuracy measure, the Mean Reciprocal Precision (MAP) and Mean Reciprocal Rank (MRR) and accuracy at level 1 (p @ 1) as estimates are employed.

[C] Question Answering Systems with Deep Learning-Based Symbolic Processing

Using the knowledge platforms described in Prolog, the authors of [C] trained the models, developed question-and-answer systems, and evaluated their effectiveness using two types of knowledge platforms with graph structures.
[D] Augmenting Dialogue Response Generation with Unstructured Textual Knowledge

In this research the authors use two evaluative measurements to measure response generation accuracy. One, the embedding average metric (EA cosine) which is negated as the cosine similarity between the embedding vectors of the sample response and the ground true response. The embedding vector of a text is the average of the word embedding in the pronunciation. This is a metric to promote semantic similarity. The other is the BLEU, which calculates the number of matches by comparing the n-gram reference of the generated answers with the n-gram of the answers. Competitions are level independent. Both of these measurements are widely accepted for measuring text similarity.

5.6 EVALUATION METRIC

BLEU (bilingual evaluation understudy) is a method of evaluating the quality of text that has been machine-translated from one natural language into another. Quality is considered to be the correspondence between the output of a machine and the human. BLEU works with the concept that it is better if the machine translation is closer to a human translation. BLEU is one of the oldest measuring metrics and it claims to have high similarity to human judgement. It is an inexpensive and popular metric.

The working of BLEU is simple, it compares the translated sentence with a quality reference translation and then a score is generated. The average overall score of the entire dataset is then generated which gives the rating of the translation. Intelligence or grammatical correctness is not taken into account. The output of BLEU is always between 0 and 1. The values, 0 and 1, refer to the closeness of the translation text to the reference text and if the translation matches exactly to the reference then the score will be 1. For this reason, it is not necessary to get a score of 1, because adding more reference translations will increase the BLEU score as there are more chances to match.

6. RESULTS AND CONCLUSION

The accuracy of the Chatbot is Measured using the BLEU measuring metric. The results obtained are compared with the baselines explained in section 5.5. The result is summarized in table 2.

MODEL	MEASURING METRIC	RESULT	ACCURACY PERCENTAGE
[A]	BLEU	0.01	50%
[B]	Rn@k	0.94	88%
[C]	BLEU	0.49	82.3%
[D]	BLEU	0.38	64.1%
ACEbot	BLEU	0.67	85%

Table 2
ACEbot BLEU result comparison

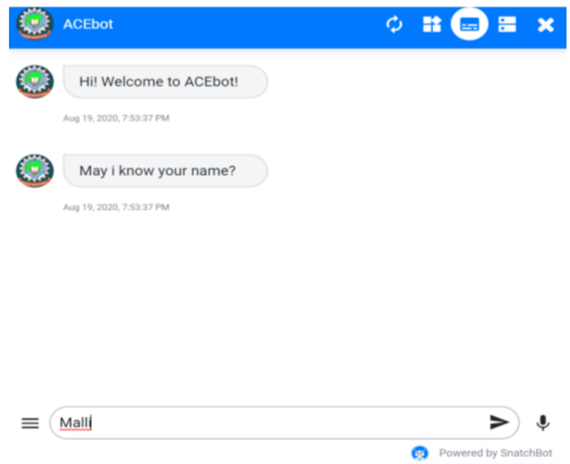
Model [A] employs Seq2seq models where the hidden level of the encoder output can be directly accessed by the decoder. This is possible due to the introduction of a focusing mechanism that uses 3 layers of stacked GRU cells for RNNs. This did not generate the expected results and thus its accuracy score is lower than the proposed system. Model [C] uses a prolog-like structure that contains a database of facts and rules which helps in answering the questions received by the system. In order to improve the learning capability of the machine, Model [C] combined Neural Machine Translation (NMT) and Word2Vec. Model [D] uses an HRED (Hierarchical Recurrent Encoder-Decoder) as a base structure. An HRED consists of three RNNs, namely a word encoder RNN, a context encoder and a decoder RNN. But Model [D] has low rate of performance, thus, its score is lower than expected. These models do not use natural language processing, which is a very powerful chatbot technology. These models did not use Natural Language Processing which is one of the most powerful chatbot technology. The baseline models [A], [C], [D] yielded lower accuracy than the proposed model because the proposed model uses Natural Language Processing along with a SEQ2SEQ model which provides a highly powerful pattern matching and extraction, and a clean dataset that generates more accurate responses without any noisy data. Using RNN model along with a more elaborate dataset will provide more accuracy to the proposed model but that will be explored in our future work. The baseline model [B] yielded high results because the authors of the model used HAMR (hierarchical aggregation network for multi-representations). HAMR is a very powerful model that consists of three layers. In the first layer, each text in the context and the candidate's response are marked by the corresponding word embedding. Two layers of bidirectional GRU are used to form the syntax and semantic representations of each sentence. A self-integration mechanism combines representations of different granularities with dynamic weights, which significantly enhances information and reduces noise.

Chatbots have evolved from being a software that provides some pass-time fun with generated answers to one of the most sought-after technology by all service-related businesses. Chatbot has significantly reduced the workload for service centres by answering many basic queries. Universities deal with a high number of prospective applicants as well as registered students who are curious about every little detail of the organisation. Satisfying the curious mind is one of the most important job of a university or college. Every minute counts, thus fast responses are necessary to improve the application average of the organisation and is made possible by chatbots. Chatbots have another important purpose and that is to collect the data. The data collected from all the enquiries helps the organisation to understand the students needs and thus, improve itself to attract more prospective applicants.

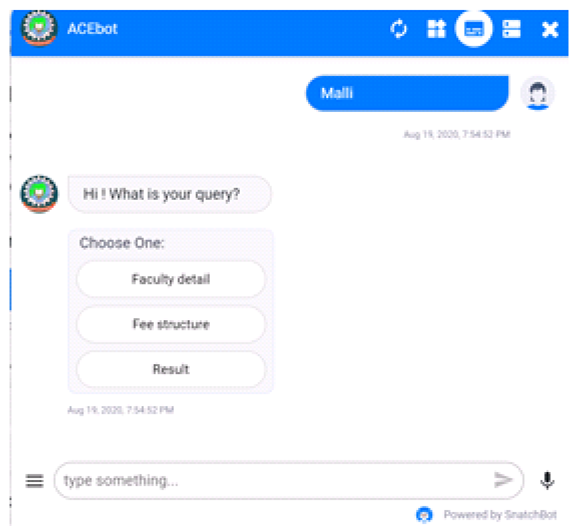
The proposed system provides an easy, interactive, and informative chatbot for the university which can be used by students currently studying in the university and students who are considering the university. The chatbot has fast response rate and also has a personality which makes it seem more human. The chatbot is trained with all the information regarding the college and their faculty. The proposed system is embedded in the university website for student access. The interaction is monitored to fine-tune the chatbot.

7. SCREENSHOTS

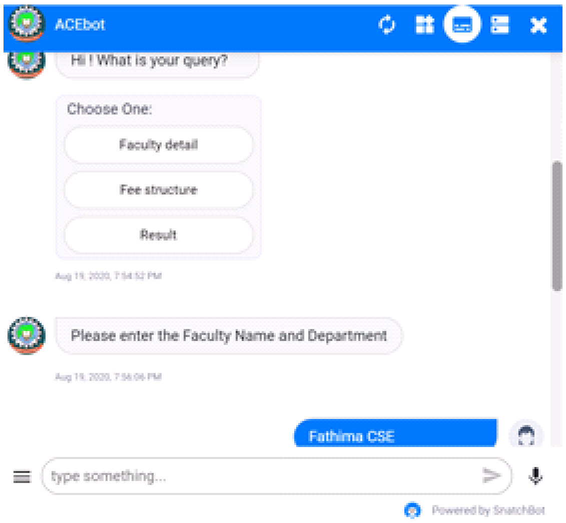
The university chatbot created provided the following results.



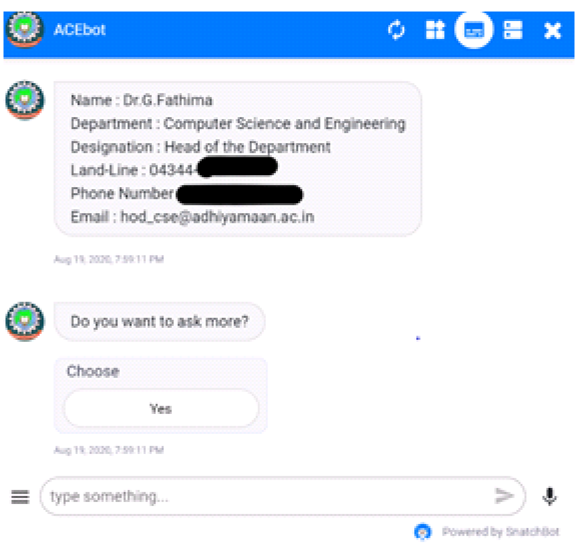
The above figure represents the welcome message of the chatbot. The welcome message consists of a formal greeting by ACEbot and enquiry of the username.



The above figure represents the response of the ACEbot to the username. Once the user gives their name, the chatbot gives the user some common query options that consists of commonly asked questions. This is possible by using the card function of the platform



The above figure represents the result of choosing the first option "Faculty detail". When the user clicks on the button an appropriate response is generated. Here, the bot asks for the faculty name and their department.



The above figure represents the result of the query. The contact information of the faculty is displayed. Then the user can either terminate or continue their query. Here, the faculty name and department are extracted from the user input. These extracted data are then matched with the keywords to display the appropriate result.

REFERENCES

[1] Huyen Nguyen, David Morales, Tessera Chin, "A Neural Chatbot with Personality" arXiv preprint arXiv:1409.0473 (2014).

[2] Guanwen Mao, Jindian Su, Shanshan Yu, Da Luo, "Multi-Turn Response Selection for Chatbots With Hierarchical Aggregation Network of Multi-Representation", IEEE Access, vol.7, 9 aug 2019, pg. 111736 – 111745.

[3] Hiroshi Honda, Masafumi Hagiwara, "Question Answering Systems with Deep Learning-Based Symbolic Processing" IEEE Access, vol.7, 17 oct 2019, pg 152368 – 152378.

[4] Lei Su, Ting He, Zhengyu Fan, Yin Zhang, Mohsen Guizani, "Answer Acquisition for Knowledge Base Question Answering Systems Based on Dynamic Memory Network", IEEE Access, vol. 7, 30 oct 2019, pg. 161329 – 161339.

[5] Yanmeng Wang, Wenge Rong, Yuanxin Ouyang, Zhang, "Augmenting Dialogue Response Generation with Unstructured Textual Knowledge", IEEE Access, vol.7, 12 mar 2019, pg. 34954 – 34963.

[6] R. E. Banchs and H. Li, ``IRIS: A chat-oriented dialogue system based on the vector space model,'' in Proc. Syst. Demonstrations Assoc. Comput. Linguistics (ACL), 2012, pp. 37_42.

[7] H. Wang, Z. Lu, H. Li, and E. Chen, ``A dataset for research on short-text conversations,'' in Proc. EMNLP, 2013, pp. 935_945.

[8] N. Kalchbrenner and P. Blunsom, ``Recurrent continuous translation models,'' in Proc. EMNLP, vol. 3, no. 39, 2013, pp. 1700_1709.

[9] K. Cho et al. (2014). ``Learning phrase representations using RNN encoder-decoder for statistical machine translation.'' [Online]. Available: https://arxiv.org/abs/1406.1078.

[10] J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan. (2015). ``A diversity-promoting objective function for neural conversation models.'' [Online]. Available: https://arxiv.org/abs/1510.03055

[11] J. Li, W. Monroe, A. Ritter, M. Galley, J. Gao, and D. Jurafsky. (2016). ``Deep reinforcement learning for dialogue generation.''[Online]. Available: https://arxiv.org/abs/1606.01541

[12] V. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau, ``Building end-to-end dialogue systems using generative hierarchical neural network models,'' in Proc. AAAI Conf. Artif. Intell., 2016, pp. 3776_3784.

[13] L. Mou, Y. Song, R. Yan, G. Li, L. Zhang, and Z. Jin. (2016). ``Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation.'' [Online]. Available: https://arxiv.org/abs/1607.00970