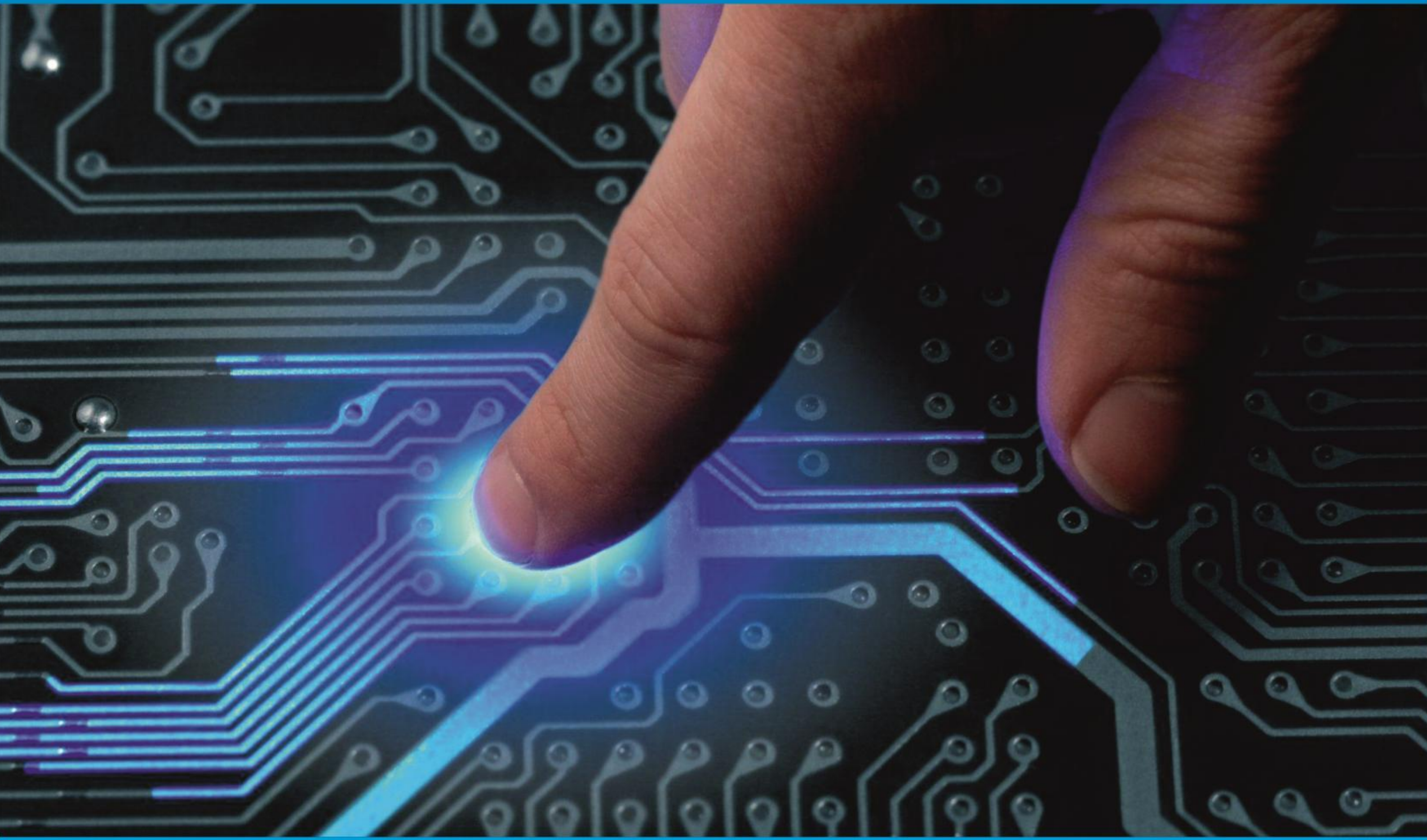
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 9, Issue 4, April 2021

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.488

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.488 |

|| Volume 9, Issue 4, April 2021 ||

| DOI: 10.15680/IJIRCCE.2021. 0904051 |

Chatbot Testing Using AI Planning

Akash Padwal[1], Pranav Enpure[2], Omkar Gujar[3], Sadanand Jagtap[4], Prof.Mrs.Susha Adagale[5]

Students, Department of Computer Engineering , Trinity Academy of Engineering, Pune,Maharashtra, India[1,2,3,4]

Lecturer, Department of Computer Engineering, Trinity Academy of Engineering, Pune, Maharashtra, India[5]

ABSTRACT: A chatbot delivers data-driven results, helping the customer service representative solve problems swifter, saving time and increasing customer satisfaction. Best of all, because the chatbot learns over time, the process will get faster as the chatbot faces the same type of calls. The purpose of this system is to display all the characteristics of "Chatbot . The main objective of preparing this document is to give a detailed description of an artificially intelligent chatbot deployed on extensively used messaging platform- Facebook Messenger, the services provided by it and how it optimizes the usage of various applications by merging it onto one platform without the hustle of switching between the apps, saving the memory, time and moreover providing rich information and performing activities with relative ease to enhance the user experience.

KEYWORDS: Chatbot ,AI, data-driven, Quick retrieval, Artificially Intelligent, Facebook Messenger.

I. Introduction

The use of virtual assistants, or chatbots , as bases for system interactions with humans is becoming of growing interests. This is due to the fact that human languages offer a natural way of communication and the availability of methods and tools allowing computers to analyze and process such languages. There are studies, e.g. predicting a rise of the chatbot market in the future. Natural language interfaces (NLI) offer a lot of new possibilities for humans to interact and collaborate with chatbots . Chatbots rely often on a set of rules or trained neural networks for deciding answers to be given to users or questions to be stated in order to handle requests. In principle chatbots are not restricted to an application domain. However, it is often the cases that chatbots rely on pre-specified patterns that trigger the chatbot's behavior, which restricts the possibilities of interaction patterns with human users. This is certainly not a drawback for applications like chatbot-based systems especially designed for a particular purpose, e.g., for providing tourism information, planning trips, or booking hotels.

With increasing complexity, chatbots can be expected to support humans in decision making. Ensuring correct functionality becomes critical for such systems. The failure to fulfil this requirement can lead to negative consequences for individuals and companies alike. Therefore, testing the chatbot's functionality will become an important task in the future.In this system, we do not focus on the underlying techniques and methods behind successful chatbot implementation. Instead, we discuss the validation and verification part behind chatbots. In particular, we are interested in providing methods and tools for automating the test of chatbots in order to increase quality of the chatbot implementation. Testing chatbots requires to provide test cases and a test execution framework for checking functional and non-functional requirements. The former is for confirming that the chatbot's behavior follows given expectations. For example, when we want to book a hotel in Prague, a hotel-booking chatbot should be able to guide the user until a hotel in Prague can be booked. The latter deals with issues like security, which is important especially in the case of textual interfaces provided over the Internet. There chatbots might be vulnerable against classical attack patterns like cross-side scripting (XSS) or SQL injection.

II. Problem Statement

The existing approval-based systems for requests rendered it difficult to handle higher numbers of transactions and larger volumes of data resulting in delays in approvals and decreased employee satisfaction. The customer needed a smart Artificial Intelligence (AI) solution which uses advanced decision-making and machine learning to not only resolve this but also customize the process as per the request while also reducing the number of inputs by the user.

III. Literature Review

1] Chatbot-based Tourist Recommendations Using Model-based Reasoning. Iulia Nica and Oliver A. Tazl and Franz Wotawa

In this paper, we present the underlying methods and technologies behind a Chatbot for e-tourism that allows people textually communicate with the purpose of booking hotels, planning trips, and asking for interesting sights worth being visit. In particular, we show how model-based reasoning can be used for enhancing user experience during a chat, e.g., in cases where too many possible selections are available or where user preferences are too restricted causing inconsistencies and as a consequence not possible answers to be provided. Besides the underlying foundations, we provide a use case from the intended tourism domain to show how such a model-based chatbot effectively can be used in practice.

2] Mohammed Javed, P. Nagabhushan, B.B. Chaudhari, "A Direct Approach for Word and Character Segmentation in Run-Length Compressed Documents with an Application to Word Spotting", 13th International Conference on Document Analysis and Recognition (ICDAR), 2015.

Mohammed Javed et al. [1] [2015]explained a method to implement word segmentation. He proposed in his algorithm to calculate character spaces in the sentences. The character spaces should include all types of gaps between characters.

They include the gaps between letter, punctuations and the words. The algorithm functions on the basis of the amount of gap or character space between each unit in the sentence. After the calculation of character spaces, an average of the gaps is calculated to know the mean average between characters in the sentence. This average gap distance is then applied to the sentence which is to be segmented. The places at which the character space is more than the average character space are said to be the points oftokenization. The gap between words is always more than the average gap and hence tokenization takes place at the blank spaces between words in the sentences.

3] Naeun Lee, Kirak Kim, Taeseon Yoon, "Implementation of Robot Journalism by Programming Custombot using Tokenization and Custom Tagging", 2017.

Naeun Lee et al. [2] [2017]proposed the implementation of word segmentation using NLTK. Natural Language ToolKit (NLTK) is a python package which caters to provide services for NLP. It has inbuilt tokenizers. Users need to import the package and use the required type of tokenizer which is present in the form of functions. The NLTK includes a wide range of tokenizers which are as follows standard, letter, word, classic, lowercase, N-gram, pattern, keyword, path, etc. The most commonly used tokenizer is the word-punkttokenizer which splits the sentences at the blank spaces. The accuracy, speed and efficiency of the NLTK tokenizers is commendable. Also, it does not require any algorithm implementation as the package executes them at the backend.

4] Tao Jiang, Hongzhi Yu, Yangkyi Jam, "Tibetan Word Segmentation Systems based on Conditional Random Fields", 2011.

Tao Jaing [3] [2011]explains the usage of CRF (Conditional Random Fields) Algorithm for word segmentation. This algorithm trains the system for spaces between the characters. Using this training, the system identifies the gap between characters in the test sentence. The system keeps a threshold value for the gap distance. If the value of gaps in the test sentence is more than the specified threshold, then the sentence splits at those points. CRF requires a lot of training to be given to the system, which makes the process time consuming. Comparing the three methods illustrated above, the NLTK proves to be more efficient in all aspects as compared to the other two. The usage of NLTK does not require the implementation of any algorithm as everything is taken care by the package itself. Also, the accuracy, speed and diversity provided by the package is better than the two algorithms.

5] Jerome r. Bellagarda, "Parts-Of-Speech tagging byLatent Analogy", IEEE Journal of Selected Topics in Signal Processing, Vol. 4, No. 6, 2010.

Jerome R. Bellegarda [4] [2010] proposed a method called latent analogy for POS Tagging. In this algorithm, latent semantic mapping (LSM) technique is used. It requires the training on the available corpus. The LSM maintains a featurespace of the trained corpus which has been tagged. Now, new sentences are provided to the LSM for tagging and the analysis is performed so as to determine the sentences from the training data which are closest to the test sentence. This is called as sentence neighbourhood. Sentence neighbourhood holds true for two sentences if they share the same intent matter. Once the intent matching sentences are found from the trained data, the POS tags attached to those sentences are then mapped to the test sentences.

eq.(3)

IV. METHODOLOGY

Java chatbot example using aiml library In this Java AIML tutorial, we will learn to create simple chatbot program in Java. A Chatbot is an application designed to simulate the conversation with human users, especially over the Internet.

Internally it uses any NLP (Natural Language Processing) system to interpret the human interactions and reply back with meaningful information.

AIML (Artificial Intelligence Markup Language) is an XML dialect for creating natural language software agents. It contains the basic rules which Natural Language Understanding (NLU) unit uses internally. It can be think of as a heart of the engine. The more rules we add in AIML – the more intelligent our Chatbot will be.

It's important to know the difference between NLP and NLU. NLP refers to all systems that work together to handle end-to-end interactions between machines and humans in the preferred language of the human. In other words, NLP lets people and machines talk to each other "naturally". NLU is actually a subset of the wider world of NLP. It helps in parsing unstructured inputs e.g. mispronunciations, swapped words, contractions, colloquialisms, and other quirks.

Proposed system:

The tested chatbot is meant for tourism purposes. As such, it offers the possibility to book a room by interacting with a user. The chatbot was developed in the known chatbot development tool, Dialogflow . The typical program flow of Dialogflow, as shown in Figure 1, consists of several steps.

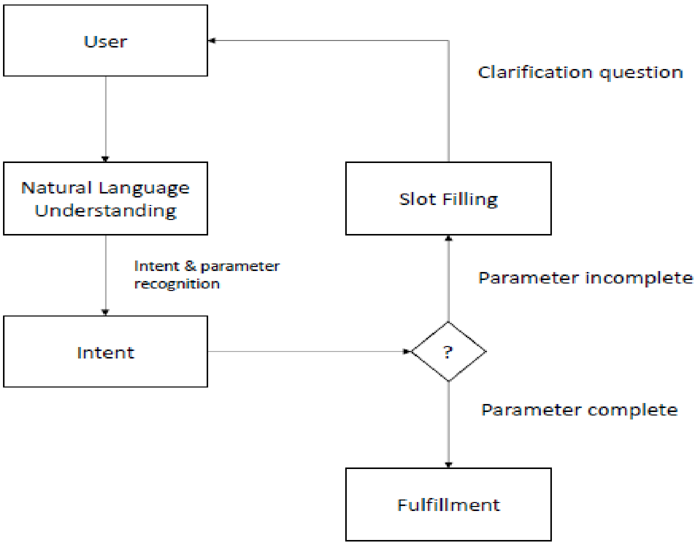


Fig. 1: System Architecture

First, the user provides an input to the system, which is then parsed by the agent, thereby responding accordingly. To specify the conversation flow, intents are used in order to classify requests. Each intent is composed of possible user inputs that can trigger this intent. It also defines what data to extract and in what manner to respond. Generally speaking, an intent represents a specific dialogue part within a conversation. In our case, the chatbot is able to recognize a booking request and respond appropriately.

Training phrases represent an important part of the intent matching of user inputs. These comprehend libraries of possible utterances, which the user could say to match the intent. Also, phrases can be annotated to match entities. These represent categories of things relevant for the intent, like city names or numbers. A typical chatbot consists of several intents. Each of them representing a variety of user intentions.

V. Conclusion

We addressed an upcoming issue in artificial intelligence, namely chatbots. These service-oriented implementations communicate in natural language and need to guarantee functionality. For this reason, we introduced a planning-based testing approach for functional testing of chatbots. Here planning is used to create a model, which serves for test case generation. The test purpose is to test a tourism chatbot by booking a hotel reservation. Planning generates sequences of actions that depict possible test scenarios. In order to refine the test generation, specific conditions can be set in planning so more diverse test cases are generated. Afterwards, a testing framework executes the test cases in an automated manner. which refers to sustaining the new system is very important and this is where most 5S systems fail. This task has been assigned to one of the team members who will look after maintenance and sustenance of the file retrieval system. Thus it can be concluded that the newly introduced 5S system is a success in many aspects.

VI. Acknowledgment

I would like to thank Prof. Mrs. SushaAdagale and the entire team of Trinity Academy Of engineering in supporting me to carry out this activity. We also take this opportunity to thank Dr. Mrs. Nikita Kulkarni from Trinity Academy Of engineering for his valuable inputs to this paper and Dr. Mr. Nilesh Uke for his support in the process.

References

[1] M. L. Mauldin, “ChatterBots, TinyMuds and the Turing Test: Entering the Loebner Prize Competition,” in *AAAI ’94 Proceedings of the twelfthnational conference on Artificial intelligence (vol. 1)*, 1994, pp. 16–21.

[2] “Chatbot Market Size And Share Analysis, Industry Report, 2014 - 2025,” https://www.grandviewresearch.com/industry-analysis/ chatbot-market, accessed: 2018-05-07.

[3] “Gartner Top Strategic Predictions for 2018 and Beyond,” https://www.gartner.com/smarterwithgartner/ gartner-top-strategic-predictions-for-2018-and-beyond/, accessed: 2018-05-07.

[4] A. Folstad and P. B. Brandtzag, “Chatbots and the new world of HCI,” *interactions*, vol. 24, no. 4, pp. 38–42, Jun. 2017. [Online]. Available: http://doi.acm.org/10.1145/3085558

[5] R. S. Wallace, “The Elements of AIML Style,” in *ALICE A.I. Foundation*, 2003.

[6] B. A. Shawar and E. Atwell, “Using corpora in machine-learning chatbot systems,” in *International Journal of Corpus Linguistics, vol. 10*, 2005.

[7] M. Utting, A. Pretschner, and B. Legeard, “A Taxonomy of Model- Based Testing,” in *Department of Computer Science, The University ofWaikato, New Zealand, Tech. Rep, 4,* 2006.

[8] A. Takanen, J. DeMott, and C. Miller, “Fuzzing for Software Security Testing and Quality Assurance,” in *Artech House, Inc., Norwood, USA*, 2008.

[9] A. E. Howe, A. von Mayrhauser, and R. T. Mraz, “Test Case Generation as an AI Planning Problem,” in *Automated Software Engineering, 4,* 1997, pp. 77–106.

[10] A. M. Memon, M. E. Pollack, and M. L. Soffa, “A Planning-based Approach to GUI Testing,” in *Proceedings of the 13th InternationalSoftware / Internet Quality Week (QW’00)*, 2000.

[11] J. Bozic and F. Wotawa, “Security Testing for Chatbots,” in *Proceedings of the 30th IFIP WG 6.1 International Conference on Testing Software and Systems (ICTSS’18)*, October 2018.

[12] ——, “Planning-based Security Testing for Chatbots,” in *Proceedings of the 21st International Multiconference Information Society - IS 2018, AS-IT-IC Workshop*, 2018.

[13] “Dialogflow,” https://dialogflow.com/, accessed: 2018-12-11.

[14] R. E. Fikes and N. J. Nilsson, “STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving,” in *ArtificialIntelligence*, 1971, pp. 189–208.

[15] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentic Hall, 1995.

[16] M. Helmert, “The Fast Downward Planning System,” in *Journal of Artificial Intelligence Research 26 (2006):191-246,* 2006.

[17] M. Katz, S. Sohrabi, O. Udrea, and D. Winterer, “A Novel Iterative Approach to Top-k Planning,” in *Proceedings of the Twenty-Eighth International Conference on Automated Planning and Scheduling(ICAPS’18)*, 2018.

INNO SPACE

SJIF Scientific Journal Impact Factor

Impact Factor:
7.488

INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

निस्केयर
NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details