

Twilio Integration with Dialogflow for Effective Communication

Sujana A S^{1}, D R Ramesh Babu¹, S Venkatesan²*

¹PG Student, ²Professor

*^{1,2}Department of Computer Science Engineering, Dayananda Sagar College of Engineering,
Bengaluru, Karnataka, India*

*Email: *sujanareddy1312@gmail.com*

DOI:

Abstract

Chatbot is gaining popularity in business groups as they will decrease customer service cost and can process numerous users' requests at a time. Thus, messaging applications are presently the utmost popular way which is used by companies to deliver chatbot experiences to consumers. But to achieve numerous tasks there is a wide requirement to make chatbots as effective as likely. To address this problem, the design of a Twilio chatbot that is integrated with Dialogflow is explained, which provides an effective and precise answer for any query based on intents and utterances configured in the Dialogflow. Chatbots can be implemented using natural language tools that are available. Dialogflow and Microsoft Luis are the two tools that are available to build a chatbot. Dialogflow (API.AI) allows developers to make conversational experiences by giving them with necessary tools to take on two kinds of tasks: first is to recognize and produce natural language utterances and second is to achieve the conversation. While there are numerous competing tools accessible in the market, Dialogflow was one of the first and continuously evolving; moreover, it is easy to use and it can be integrated with other services.

Keywords: *Chatbot, Dialogflow, Microsoft Luis, natural language tools, twilio, utterances*

INTRODUCTION

Chatbot is a program that interacts with us. It is a layer on top of the software services. The services that a chatbot gives are various. Significant life-saving health messages, to check the weather forecast or to obtain a new pair of shoes, and whatever else in between. The term chatbot is synonymous with the text conversation but is rising quickly via voice communication "Alexa, what is the time?".

Twilio is an API used for effective communications. Software teams use Twilio APIs to enhance abilities like voice, video, and messaging to their applications. This permits businesses to deliver the right communications experience for their customers. Behind Twilio APIs is a super network, a software layer that ties and enhances communications networks everywhere the world. This is what allows

operators to consistently call and message anyone anywhere. With Twilio, you can contact customers in the ways they prefer, and engage with them efficiently using context related to that communication. As customer experience can gradually make or break your brand, programmable communications have become more critical than ever to the accomplishment of businesses today. The chatbot can talk to us through different channels; such as Facebook Messenger, Siri, WeChat, Telegram, SMS, Slack, Skype and many others. Consumers spend lots of time using messaging applications (more than they spend on social media). Therefore, messaging applications are currently the most popular way companies deliver chatbot experiences to consumers.

Dialogflow is a tool which provides a way to design human-computer interaction

based on natural language conversations. It lets app developers to deliver their users with original ways of interacting with their interfaces via voice and text exchanges driven by artificial intelligence. With Dialogflow, developers have permission to integral machine learning and natural language processing technologies. This lets them to emphasis on other integral

parts of app creation rather than on defining in-depth grammar rules. Dialogflow has pre-built agents that will let users build their applications earlier. On top of that, the restored API.ai has an in-line code editor so that coders can achieve numerous tasks straight from the console.

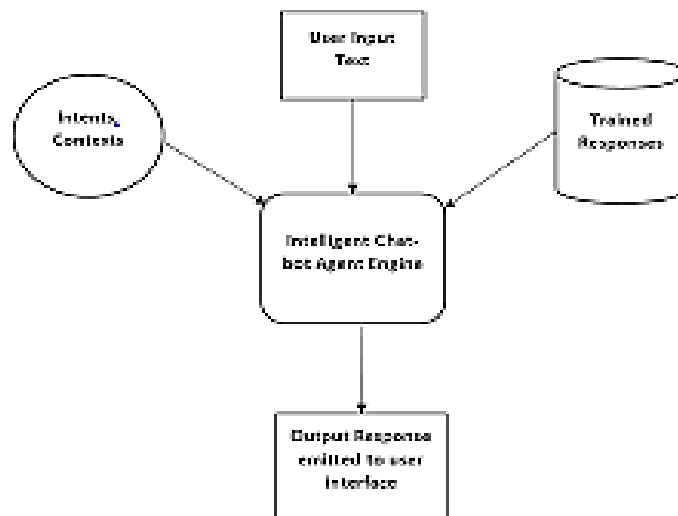


Figure 1: Dialogflow architecture.

Once developer created intents and utterances for customers query there will be an option on the console to integrate. Integration option lists various API'S to integrate with dialogflow like Facebook, Instagram, whatsapp, Twilio and so on. On clicking selected API, Dialogflow will be integrated with respected API. Generally, user conversation as a rule starts with welcome or general queries. First user sends his/her text from Twilio number. Text that is given by user will be sent to the interactive chat bot agent engine of Dialogflow where user message will be processed and searched for related intents and contexts and then trained response will be sent to the user. This is how Twilio can be integrated with Dialogflow.

RELATED WORK

The e-business has totally altered the way of selling products. E-commerce is one of

the best e-business models which frequently do business over the internet. The major disadvantage of this field is quality of customer service they deliver. In every e-business model, customers have to pause for a long time to get response from the customer service representative. Particularly in case of live chat, they interact with numerous customers at a time. The responses may not be appropriate as they will copy paste pre-written responses. Also, the slow response and the long time wait time is the main drawback in online services. As an answer to this problem, author designed a chatbot which mechanically gives straight responses to the users based on the data set of Frequently Answered Questions (FAQs), using Artificial Intelligence Markup Language (AIML) and Latent Semantic Analysis (LSA). Template based queries like general questions and greetings are

answered using AIML and other service associated questions use LSA to give replies. Author proposed a chatbot in such a way that for solo template, it will deliver random replies. LSA based questions are giving right replies. In future, the chatbot will be clever to answer general queries significantly by giving more templates and patterns [1].

Successful application of a chatbot calls for accurate analysis of user's query by the bot and the formation of the precise response that should be given to the user. In many scenarios the evidence available from the user's query is insufficient to provide the answer. In such cases, the chatbot needs to be intrusive so that it will be communicating and can mimic a natural human interaction. This paper reports the employment of an intrusive chatbot, which will discover the missing data in query and enquires the queries to users to collect data that are required to answer the query. Over this implementation, the level of interactivity among the user and the chatbot will be better. The proposed system is implementing an extra knowledge base engine (KB engine) to the existing system and interfaces this with a database for attractive accurate data for replying to certain queries. Since the query is first examined in the AIML, we need to have a mechanism in the AIML to teach the chat engine, to direct the query to KB engine for pointing the database [2].

INES (INtelligent Educational System) is a useful prototype of an online learning platform, which syndicates three vital abilities associated to e-learning actions. These abilities are those concerning to a LMS (Learning Management System), a LCMS (Learning Content Management System), and an ITS (Intelligent Tutoring System). To carry out all this functionality, the system, as a whole, includes a set of

dissimilar tools and technologies, as follows: semantic managing users (administrators, teachers, students...) and contents tools, an intelligent chatterbot able to connect with students in natural language, an intelligent agent founded on BDI (Believes, Desires, Intentions) technology that acts as the brain of the system, an inference engine built on JESS (a rule engine for the Java platform) and ontologies (to modulate the user, his/her activities, and the learning contents) that pay with the semantics of the system, etc. In this paper author emphasis on the chatterbot, CHARLIE (Chatter Learning Interface Entity), established and used in the platform, which is an AIML based (Artificial Intelligence Markup Language) bot. Author addressed its performance and its contribution to INES [3].

The key limits of chatbot technology are connected to the building of their information depiction and to their firm information retrieval and dialogue capabilities typically based on humble "pattern matching rules". The inspection of distributional assets of words in a texts corpus permits the formation of semantic spaces where represent and compare natural language elements. This space can be taken as a "conceptual" space where the axes indicate the latent primitive concepts of the examined corpus. The existing work in these paper goals at exploiting the properties of a data-driven semantic/conceptual space built using semi structured data sources that are easily available on the web, like Wikipedia. This coding is correspondent to adding, into the Wikipedia graph, a conceptual similarity relationship layer [4].

METHODOLOGY

The system architecture of Twilio integrated with Dialogflow is shown below.

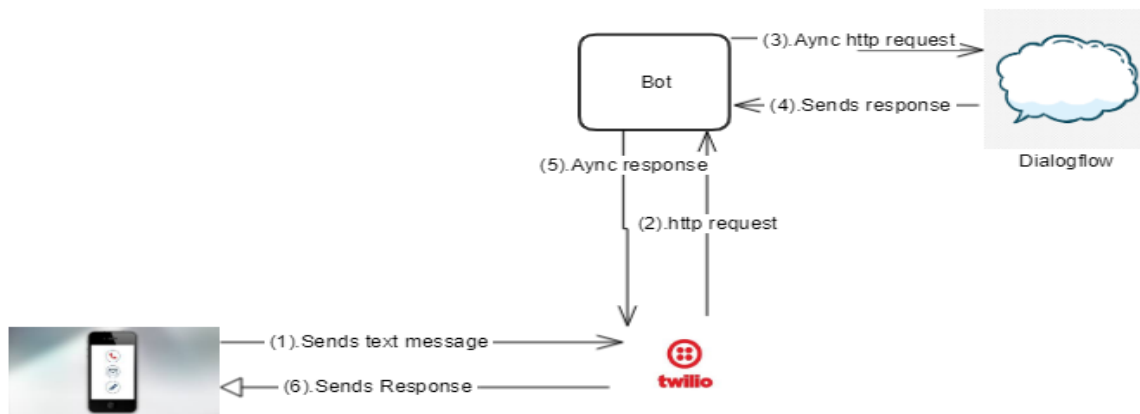


Figure 2: System architecture.

First, user sends a text/voice message from his/her Twilio phone number. Message will be sent to the interactive bot engine where message will be processed, and picks associated utterances from dialogflow. The message is characterized and harmonized to a corresponding intent (Intents are well-defined manually by creators in Dialogflow). Developer should describesubsequent actions for respective intent in the Webhook. When a particular matched intent is found by Dialogflow, the webhook will custom external APIs to discover a reply in external databases. The external databases direct back essentialdata to the webhook. Webhook directsconfiguredanswer to the intent. Intent produces actionable informationconferring to dissimilar channels. The actionable informationgoes

to output Twilio. The user gets a text/image/voice answer.In the above description, webhook is a technique of altering the performance of a web page, or web application, with custom callbacks. These callbacks can be preserved, modified, and achieved by third-party users and creators who may not essentially be associated with the originating website or application.

Setting up Dialogflow

First, create a developer account on API. AI (Dialogflow). Developer must Sign in with his/her credentials. Later after signing in, generate a basic agent. To generate a new agent, click the drop-down menu on the left side of the home page and then click Create new agent.Fill in the method on the right. Give a term and explanation. Select time zone and then click CREATE.

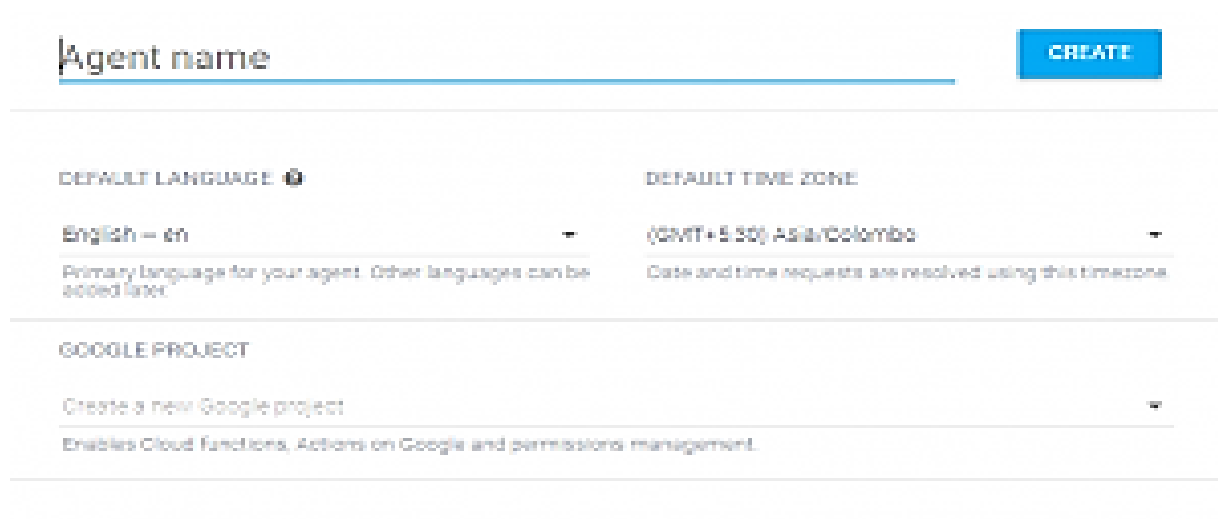


Figure 3: Agent creation in Dialogflow console.

This will take developer to the page with the intent's entry. Developer can notice that there are two intents

previously: Default Fallback Intent and Default Welcome Intent.



Figure 4: Intent page in Dialogflow console.

Developer must add his/her initial intent. Intent is what the operator or bot needs to convey using utterances or button presses. An intent is a representative of an utterance. We need intents because there are numerous ways to ask for the same thing. The procedure of recognizing intents is to map the several ways explicitly to an intent. For example, the user could ask to recognize the weather condition in their city by means of the following utterances:

"hows the weather in London"
"whats the weather like in London"
"weather in London"
"is it sunny outside just now"

In the previous utterances, the user is requesting for a weather report in the city of London. The first step of algorithm is to map these numerous utterances into a single intent. The Intent name resembles to users'

intents. So, name them from the user's viewpoint. To add an intent, click the CREATE INTENT button. Enhance sample user utterances in the User says text field. These are model utterances that will help the agent to recognize the user's intent. Now add a few greeting utterances that the user asks to Chatbot.

Hello
hello there

In the response tab, add the bot's answer to the user. This is the definite utterance that the bot will direct to the user. Developer can enhance more answers so that the agent can arbitrarily pick one to make it less tedious and boring:

Hi there. I am John. Nice to meet you!
Up to 10 extra answers can be added by clicking the ADD MESSAGE CONTENT.

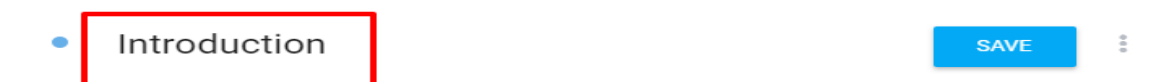


Figure 5: Saved Intent(introduction).

Later, click on SAVE button in the top-right corner to save the intent. Test it by means of the simulator on the right side of the page.

Integrating Twilio with Dialogflow
First developer should sign up and create a Twilio account. After logged in, Twilio phone number should be purchased to send

text/rich text messages. After creating a chatbot, it must be published on a platform where users can really use it. Dialogflow allows designers to integrate with

numerous platforms. Click Integrations to see all the platforms that are presented and try to deploy with Twilio or any with available API.

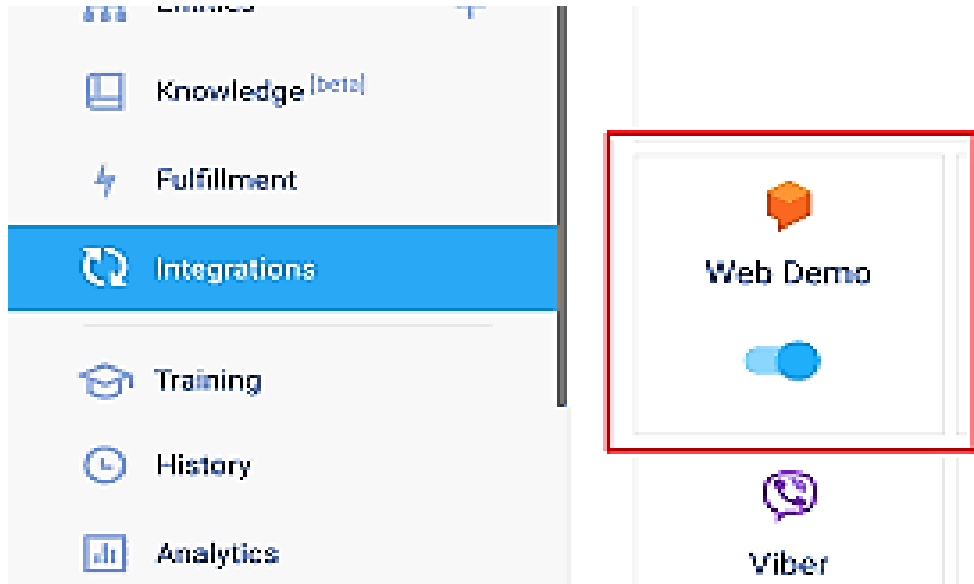


Figure 6(a): Integration menu with available API's.

Integrations

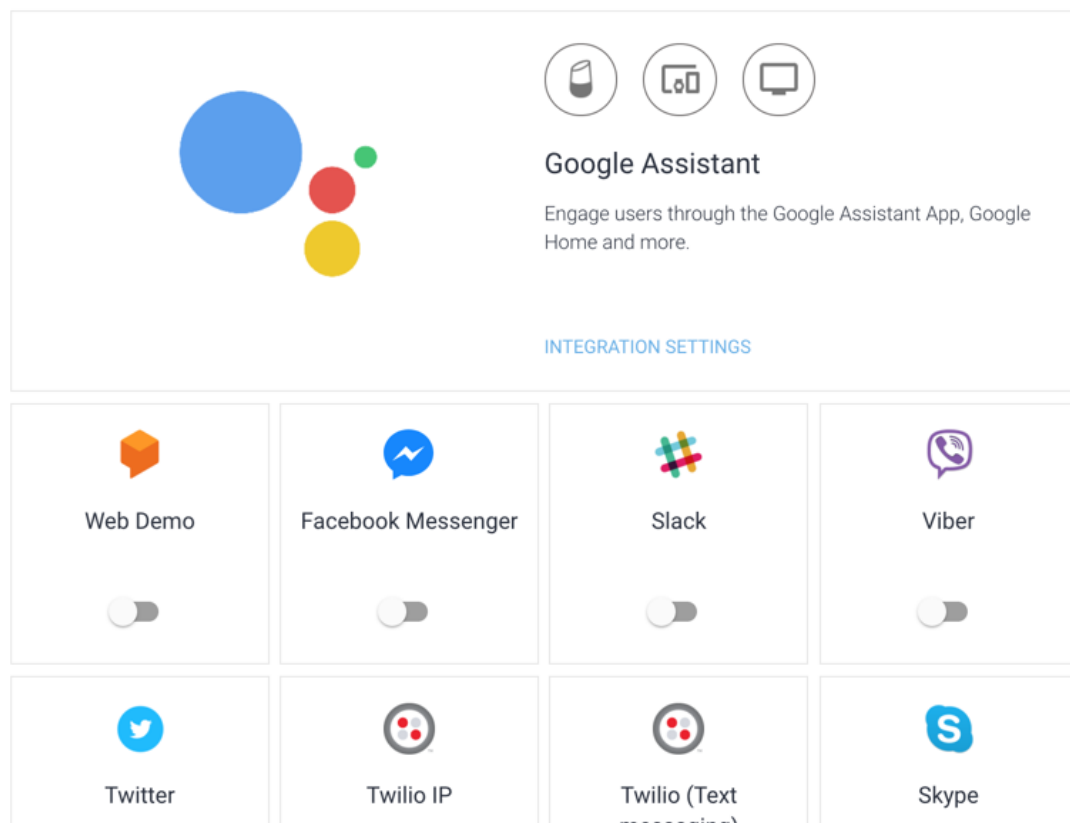


Figure 6(b): Integration of Dialogflow with Twilio API.

EXPERIMENTAL ANALYSIS

Intent classification is a significant component of any Natural Language Understanding (NLU) system in any chatbot based platform. For the chatbot to work well, it should be recognized appropriately to the intent of the user from user input in order to activate the correct action or dialog. Usually in any bot-building platform, the bot developer creates a list of intents, and for each intent provides a set of training phrases representing what a normal user may say for that intent. The amount of training phrases differs across intents and bots. One

can guess from a few (<10) to a hundred and even more training phrases per intent. Once the bot is trained, the bot's intent sorting is estimated using testing phrases to see if the bot notices the intents appropriately or not. One can compare the intent classification results in the above paper with those for Botfuel NLP classification service. The Botfuel classification service is part of Botfuel bot building platform. The below graph evaluates the performance of the Twilio chatbot integrated with Dialogflow intent classification by comparing it with other available NLP tools.

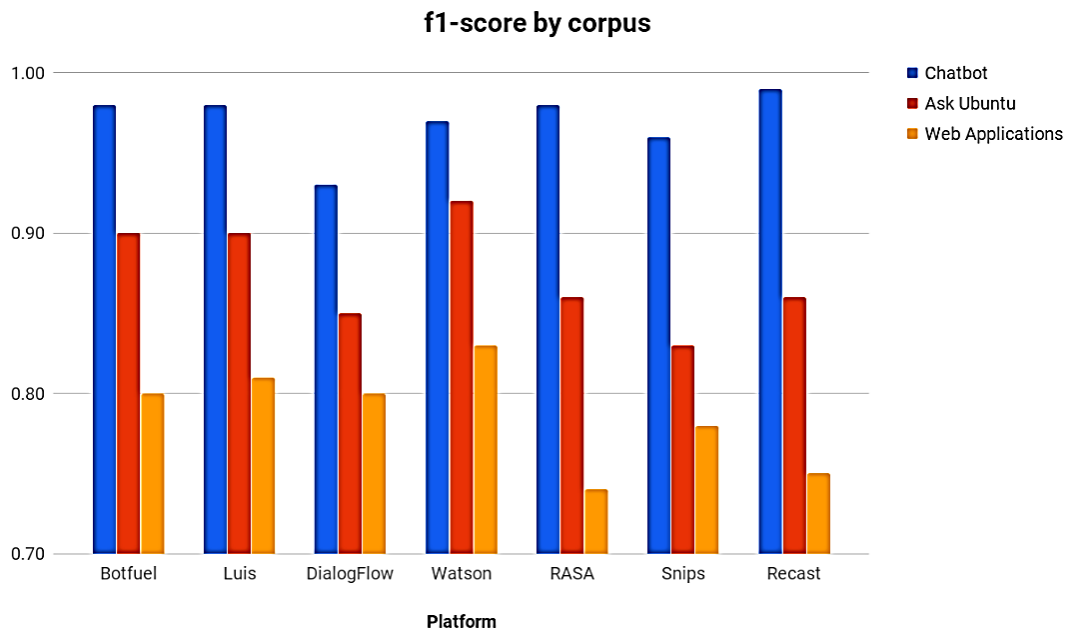


Figure 7: Comparison of Dialogflow with other available NLP tools.

CONCLUSION

It utilizes natural language in conversations. In this article integration of Twilio with Dialogflow gives immediate responses to the customer queries that are related to the problems in the network and functioning of application. It helps to solve customer related queries easily and reduces human intervention. One can send text messages, videos and images using Twilio chatbot application. Application developers are given the tools to enhance

their app's interaction features for their customers through AI-powered text and voice discussions. With Dialogflow's help, developers can emphasize their time on other applicable parts of the creative process while the platform switches the standard protocols and functionalities requiring in-depth grammar rules.

The applications developed in the system use natural language processing. Dialogflow also features an in-line editor

so that developers can write the code directly from the console.

REFERENCES

1. N. Thomas (2016), "An e-business chatbot using aiml and lsa", *Advances in Computing, Communications and Informatics (ICACCI)*, 2016 International Conference on. IEEE, pp. 2740–2742.
2. S. Reshmi, K. Balakrishnan (2016), "Implementation of an inquisitive chatbot for database supported knowledge bases," *Sadhana*, Volume 41, Issue 10, pp.1173–1178.
3. Fernando A. Mikic, Juan C. Burguillo, Martin Llamas, Daniel A. Rodriguez (2009), "CHARLIE: An AIML-based Chatterbot which Works as an Interface among INES and Humans", *EAEIE Annual Conference*, Dob10.1109/Eaeie.2009.5335493.
4. Agnese Augello, Giovanni Pilato, Giorgio Vassallo and Salvatore Gaglio (2009), "A Semantic Layer on Semi-structured Data Sources for Intuitive Chatbots", 2009 International Conference on Complex, Intelligent and Software Intensive Systems, DOB 10.1109/CISIS..165.
5. A. M. Neves, F. A. Barros, C. Hodges (2006), "Iaiml: A mechanism to treat intentionality in aimlchatterbots", *Tools with Artificial Intelligence, ICTAI'06. 18th IEEE International Conference on. IEEE*, pp.225–231.
6. M. d. G. B. Marietto, R. V. de Aguiar, G. d. O. Barbosa, W. T. Botelho, E. Pimentel, R. d. S. Franc, a, V. L. da Silva (2014), "Artificial intelligence markup language: A brief tutorial," arXiv preprint arXiv:1307.3091, (*Engineering Science*), Volume 35, Issue 6, pp. 47–51.

Cite this article as: