A Voice Based Assistant Using Google Dialogflow and Machine Learning

Dr. Jaydeep Patil[1], Ekta Bhushan[2], Rucha Khartadkar[3], Atharva Shewale[4], Alister Fernandes[5]

[1]*AISSMS Institute of Information Technology, Pune, India*
[2,3,4,5]*AISSMS Institute of Information Technology, Pune, India*
jaideep.patil@aissmsioit.org[1]
ekta1624@gmail.com[2]
rucha.khartadkar123@gmail.com[3]
shewaleatharva@gmail.com[4]
alisterfernandes24@gmail.com[5]

Abstract-

Virtual Personal Assistant (VPA) is one of the most successful results of Artificial Intelligence, which has given a new way for the human to have its work done from a machine, just like a normal person. This paper provides methodologies and concepts used in making of a Virtual Personal Assistant (VPA) and thereby going on to use it in different software applications. In this project, we are trying to make a Virtual Personal Assistant ERAA which will include the important features that could help in assisting ones' needs. Keeping in mind the user experience, we will make it as appealing as possible, just like other VPAs. Various Natural Language Understanding Platforms like IBM Watson and Google Dialogflow were studied for the same. In our project, we have used Google Dialogflow as the NLU Platform for the implementation of the software application. The User-Interface for the application is designed with the help of Flutter software platform. All the models used for this VPA will be designed in a way to work as efficiently as possible. Some of the common features which are available in most of the VPAs will be added. We will be implementing ERAA via a smartphone application, and for future scope, our aim will be to implement it on the desktop environment. The following Paper ensures to provide the methodologies used for development of the application. It provides the obtained outcomes of the features developed within the application. It shows how the available natural language understanding platforms can reduce the burden of the user, and therefore going on to develop a robust software application.

Keywords-* *Virtual Personal Assistant, Artificial Intelligence, Natural Language Understanding, IBM Watson, Google Dialogflow, Flutter

Introduction

In this new age of technology and innovation, the use of artificial intelligence and machine learning has made our life much easier. These technologies have proved to be beneficial to the society in various fields such as education, industries, e-commerce, etc.; and one of the most prominent is communication. Right from the 1960s, when IBM introduced the first digital speech recognition tool, i.e. IBM Shoebox, the idea of having a meaningful conversation with a computer seemed quite futuristic. A Personal Virtual Assistant is a digital life assistant made to contribute maximum convenience to the user. Most of the Virtual Assistants work basically on voice as communication. It focuses on processing of audio signal into the system, converting them to text and performing the required task. In general, speech processing consists the following: A Speech-To-Text Module that converts speech signals to text, A Parser that extracts the semantic context, A Dialog Manager that determines system response through machine learning algorithms, An Answer Generator that provides the system response in text and A Speech Synthesizer that converts text to the speech signal [11]. When developed by a normal user, he may experience many issues, in terms of accuracy in recognition, robustness in performing operations, etc. and at times may not be able to understand the issues faced. Therefore, in this paper, we have tried to give an overview which will help user understand the methodologies and steps involved in the making of a Basic Virtual Assistant. We have taken into consideration the different methodologies, results and limitations published by different researchers. The Application developed with the help of Google Dialogflow help the user to perform various task at ease and thereby burdens the load of user during busy schedules.

I. Literature Review

The Speech Recognition Model is one of the most important parts of a Virtual Assistant. Considering the various Neural Networks that are required for building up a speech recognition system, it was necessary to survey the models that provided the insight by determining the accuracy and other factors of each Model. It was observed that High Accuracy and less Validation Accuracy was achieved for Convolutional Neural Network (CNN) model as compared to Basic Neural Network. Thus, proving that CNN is a better choice for speech recognition systems[1]. Considering the Limitations for the Model, other parameters such as Word Error rate, throughput of the system was not taken into consideration. Various Machine Learning Algorithms are used for speech recognition. It was found that on application of Auto- WEKA on various algorithms, Random Forest was the best algorithm which is useful for learning the dataset based on the training set. However in this survey, Speech samples consisting of noise were not tested for determining the scalability and robustness of the models[3]. In the Survey of scaling speech recognition using CNN, following metrics were taken into consideration:

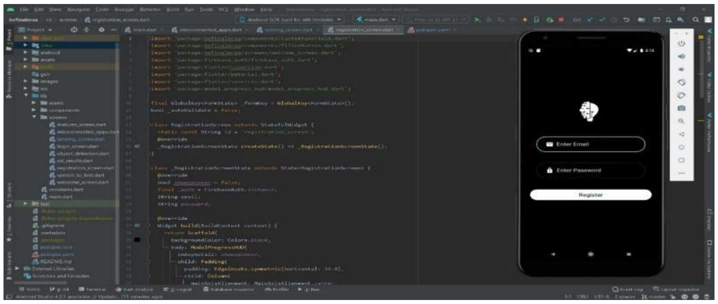
(i) throughput,
(ii) Real-Time Factor(RTF) and latency, and
(iii) Word Error Rate (WER)

for the overall framework, helped in achieving an efficient model. But due to the increase in the number of the layers the implementation of the same was difficult[4]. Other Algorithms such as the Long Short-Term Memory (LSTM) are very powerful in speech recognition and Hybrid models of Hidden Markov Model (HMM) and Gaussian Mixture Models (GMM) can give excellent results[5]. In various Projects of developing a Virtual Assistant it was observed that the platform failed to support various other languages of the countries including China, Japan, India, etc.[6]. The survey paper provided detailed study on the Recurrent Neural Networks (RNNs) that can be used for Speech Recognition System but with more research to be carried out on the same. However, the survey focused more on the Supervised Learning Models and less importance was given to the Unsupervised Learning Models. A Survey included the detailed comparison of the Personal Voice-Based Assistants available in the market namely, Google Assistant, Cortana, Alexa and Siri. It concluded that Google assistant gave good results in VR and HFI by achieving 60% accuracy. Siri achieved 44% accuracy in VR and HFI. Cortana was observed with a decrease in accuracy close to 30%. Other results included that Alexa wasn’t suitable with simple questions whereas Cortana was poor in basic voice recognition[7]. The illustration to use AI-enabled content analysis has been discussed in one of the survey papers. The system can examine text of leadership speeches, content related to a specific organization. However, Only one type of content was analyzed with limited samples and a Pre-defined Coding Scheme was used for the Project[9]. Since, the survey of this paper also included the comparison of IBM Watson and Google DialogFlow, various Projects carried on these platforms were studied before arriving at the conclusion for a better Platform for the Project. A Project was based on successful implementation of IBM Watson in developing an application for health care purposes[12]. This Project thus provided a base for building up an AI Application with the help of IBM Watson. Various other Projects used IBM Watson as their platform for building the system which processed various queries with the help of its in-built Natural Language Processing (NLP) and Natural Language Understanding (NLU) Algorithms[8]. Projects based on successful implementation of the Google DialogFlow for an Organization were studied. It provided an insight to the various technologies like the Google Cloud Platform, Google Cloud Vision API for integrating detection features in the system and Firebase RealTime Database for developing the Application. The system ensures security of the database with the help of OAuth Authentication for accessing the system. However, most of the actions carried out with the help of Google services required Internet Connectivity while accessing the system and thereby failed to service the queries offline[11]. Another Project, aimed to design a system for Educational purposes using Google DialogFlow. The proposed methodology consists of two main phases: Knowledge Abstraction and Response Generation. The methodology studied the deep learning model, The Decision Tree, that has been used in implementing the Dialogflow [10].

II. Methodology

User-Interface

The User Interface for our application was developed with the help of Android Studio and Flutter, an Open Source UI Software Development Platform. The graphics library and packages available in this platform allowed faster operation of the application. One of the designs of the page of our application is as shown below.



```
import 'package:befinaleraa/components/customTextField.dart';
import 'package:befinaleraa/components/filledButton.dart';
import 'package:befinaleraa/screens/welcome_screen.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:modal_progress_hud/modal_progress_hud.dart';

final GlobalKey<FormState> _formKey = GlobalKey<FormState>();
bool _autoValidate = false;

class RegistrationScreen extends StatefulWidget {
  static const String id = 'registration_screen';
  @override
  _RegistrationScreenState createState() => _RegistrationScreenState();
}

class _RegistrationScreenState extends State<RegistrationScreen> {
  @override
  bool showSpinner = false;
  final _auth = FirebaseAuth.instance;
  String email;
  String password;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.black,
      body: ModalProgressHUD(
        inAsyncCall: showSpinner,
        child: Padding(
          padding: EdgeInsets.symmetric(horizontal: 34.0),
          child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
```

Fig – 1 User Interface Design

It promises to give a stunning look to the Application Interface, irrespective of the operating platform. It enables the developer to create Cross-Platform Applications with ease. Thus the need of developing different applications to run on Android and iOS is eliminated. Apart from Cross-Platform feature flutter promises to provide various other features namely, Hot Reload which enables the developer to see the changes in the code instantly which is reflected on the UI as shown in the above Fig – 1, Widget Library which helps in creating complex widgets that can be customized as per requirements, Minimal Code as flutter being a Dart Programming Language it uses Just-In-Time (JIT) and Ahead-Of-Time(AOT) compilation that improves the overall startup time, functioning and accelerates the performance thus providing an efficient platform to develop an application.

Dialog Manager

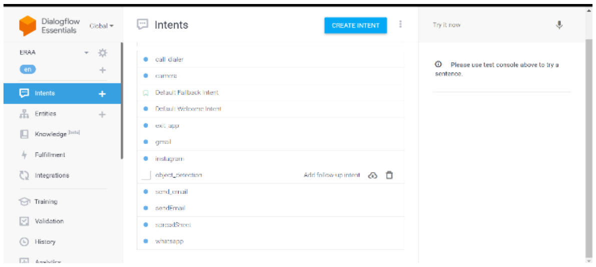
The Dialog Manager is the main component of the Virtual Assistant. It handles the entire flow of conversation between the user and the virtual assistant. The input to the dialog manager is the user expression, which is converted into system understandable language with the help of Natural Language Platform, which commands the application to perform the required task as given by the user. For developing a dialog manager for our project we used the Google Dialogflow's ES(Standard) version. This free user platform allows one to manage the conversations to be carried out between the system and the user with its various features including Natural Language Processing and Machine Learning Models. The Software also provides the Speech-To-Text API which could help in enabling speech recognition features in the system. The dialogflow learns itself and applies a machine learning model for the entire dialogflow agent. The ES version of Dialogflow provides various features. They are as follows:

Dialogflow Agent

It is responsible for handling the user conversations and converting the voice command given by the user into the text or the text command into Structured Data that is understandable to the application. Each of the Agents consists of Intents and Entities. For our application ERAA, we have created an agent ERAA in the google dialogflow console where the entire conversations and the tasks that are to be carried out by the application are defined.

Intents

It takes care of matching the user expressions obtained from the previous step to the best intent in the agent. This matching of intent is also known as Intent Classification.
Fig – 2 shows the various Intents that we created for ERAA.



- call dialer
- camera
- Default Fallback Intent
- Default Welcome Intent
- exit app
- gmail
- instagram
- object_detection
- send_email
- sendEmail
- spreadSheet
- whatsapp

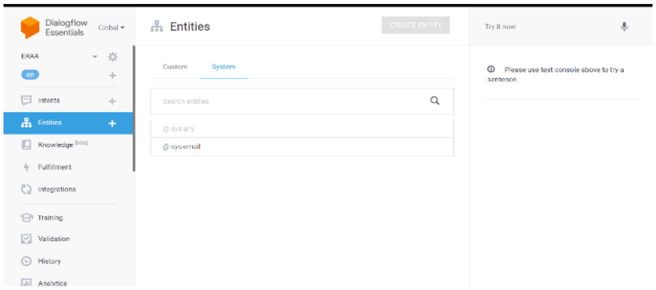
Please use test console above to try a sentence.

Fig – 2 Intents

The Intents defined are based on the tasks that ERAA can perform when invoked by the user. The Default Fallback Intent is added into the agent which will be triggered when the dialogflow is unable to interpret the command given by the user to the application and the Default Welcome Intent includes the WELCOME Event that triggers this intent at the beginning of the interaction with the dialogflow. Various other intents differ from each other depending upon the parameters, actions that are to be triggered in response to the user command.

Entities

Pre-defined System Entities are provided by Dialogflow for matching dates, times, email addresses and so on. Entities can also be user-defined depending on the type of data handled by the system application.
The following Fig – 3 shows the entities used for the application.



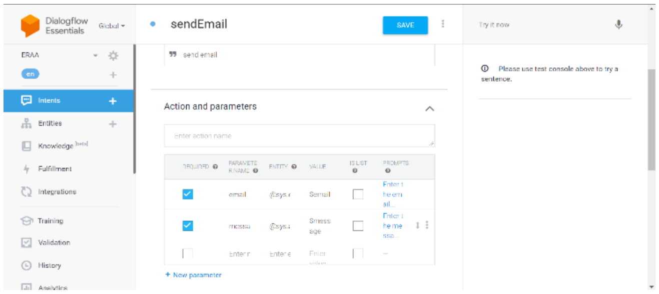
@sys.any

@sys.email

ⓘ Please use test console above to try a sentence.

Fig 3 – Entities

We have used two system entities in the agent for accomplishing the task of sending an email through the application ERAA. The System Entity ‘@sys.email’ contains the extracted email address to which the user would like to send an email to and the System Entity ‘@sys.any’ contains the extracted body of the email to be sent from the ‘sendEmail’ Intent present in the agent. The ‘sendEmail’ intent is as shown below in Fig – 4.



send email

REQUIRED	PARAMETER NAME	ENTITY	VALUE	IS LIST	PROMPTS
☑	email	@sys.e	$email	□	Enter the email...
☑	messa	@sys.a	$message	□	Enter the messa...
□	Enter r	Enter e	Enter value	□	–

ⓘ Please use test console above to try a sentence.

Fig – 4 The ‘sendEmail’ Entity

For accessing the user email address the webhook call for this intent is enabled. The Webhook Call in Google Dialogflow can access ‘https’ that is the secured http request and the URL for the requests must be publicly accessible. It is responsible for handling POST requests with a JSON WebhookRequest body and responds with the JSON WebhookResponse body.

3.2.4 User Interactions with the API

For interacting with the Dialogflow API Service, a code must be written for direct interaction. Fig-5 shows the processing flow when interacting with the API Service.

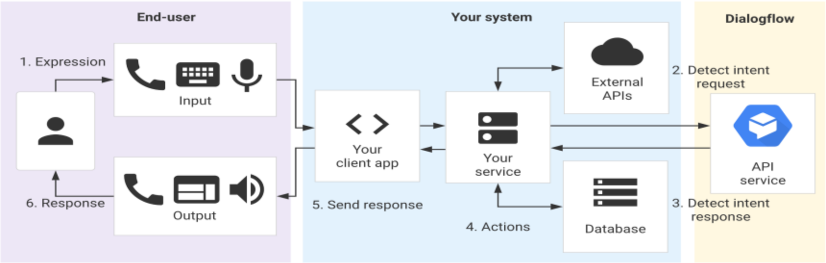


Fig – 5 Interaction with the API

a. The end-user types or speaks an expression.
b. Your service sends this end-user expression to Dialogflow in a detect intent request message.
c. Dialogflow sends a detect intent response message to your service. This message contains information about the matched intent, the action, the parameters, and the response defined for the intent.

d. Your service performs actions as needed, like database queries or external API calls.
e. Your service sends a response to the end-user.

The feature of Text-to-Speech of the system is developed by ‘flutter_tts’, a Text-To-Speech Package provided by Flutter. It helped in providing answers to the queries of the user in audio format. Thus, enhancing the usability of the Application.

Speech Recognition

The input to the application could be either in text or in voice as per the convenience of the user. In terms of voice commands the Dialogflow possessed an in-built feature of Speech-To-Text API that included various Machine Learning and Neural Network Algorithms to accomplish the extraction of text from speech even in noisy environments. Thus, the Speech Recognition was an added feature in ERAA, that made it more suitable to the users during their heavy workloads.

Other Features

The feature of opening device applications required accessing permission for the same which was handled by the ‘permission_handler’ plugin offered by Flutter. This plugin provides a cross-platform API to request and check permissions for the other applications present in the device. Thus, ERAA was then able to handle such requests from the user.

Object Detection Feature

We have implemented a pre-trained Convolutional Neural Network (CNN) Model which is created by Google Cloud. For mobile specific usage these features are available on Google Firebase. The CNN Model helps in detecting the object with higher confidence level.

The below Fig – 6 shows how CNN helps in detecting Objects. It includes various layers of the network namely Convolutional layer and the Max Pooling layer. Each layer extracts the most detectable feature from the image and converts them into a vector. A fully connected layer is developed within these vectors and then the image is detected based on its training set. This Neural Network consisting of deep layers within itself is then able to detect the images on application of the test set.

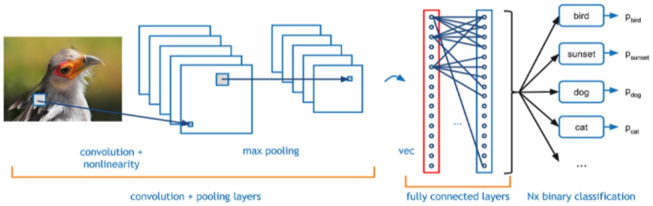


Fig – 6 Object Detection with the help of Convolutional Neural Network (CNN) Model

ISSN: 2233-7853 IJFGCN

IV. Results

We have implemented our Login and Sign-Up page using firebase which is like a database of all registered users for our application. The following Fig – 7 and Fig – 8 shows the Launch Page of ERAA and the Login/Sign-Up page of ERAA.

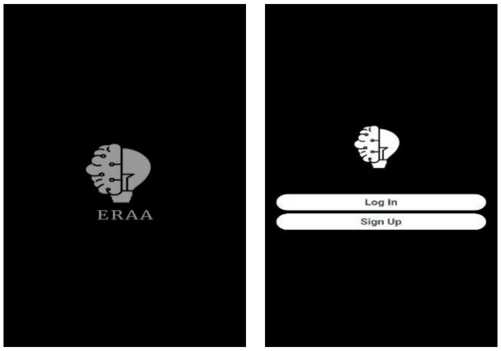
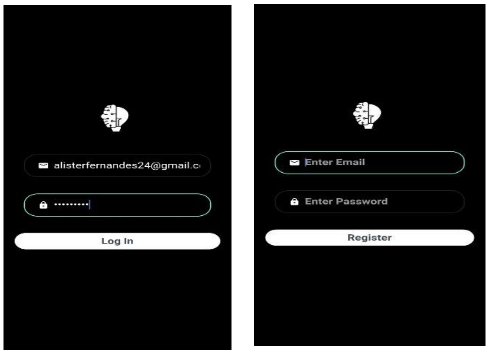


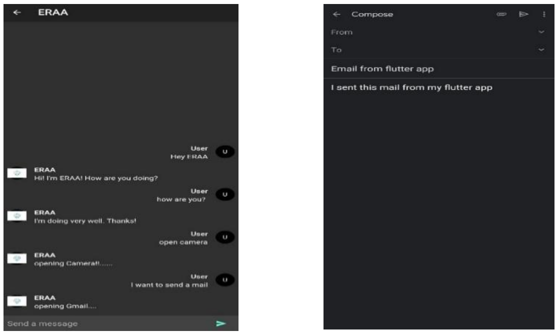
Fig – 7 Launch Page



alisterfernandes24@gmail.c

Fig – 8 Login/Sign-Up Page

After login, the user is directed to a conversation page where the user is able to interact with ERAA. The tasks that ERAA is able to perform includes small talk with user and opening of various applications present in the device. The application also provides the facility of sending an email through its own interface provided that the user has already signed up their mail account in the Gmail App present in their device. This allows the user to compose the email message and send it to the intended person at one go. The following Fig – 9 shows the same.



User
Hey ERAA

ERAA
Hi! I'm ERAA! How are you doing?

User
how are you?

ERAA
I'm doing very well. Thanks!

User
open camera

ERAA
opening Camera!!......

User
I want to send a mail

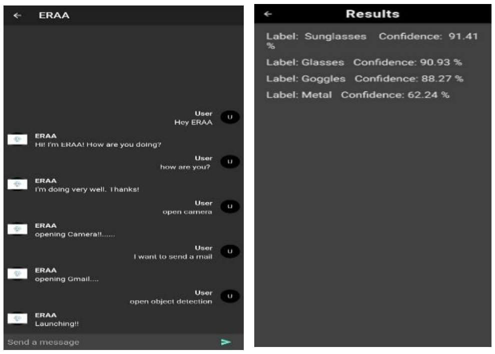
ERAA
opening Gmail....

Email from flutter app

I sent this mail from my flutter app

Fig – 9 Sending Email through ERAA

The Object Detection feature allows the user to detect any object either by choosing one from the device gallery or by opening the camera of the device and clicking the picture of an object for the same as shown in Fig – 10(a) and (b).



ERAA

User
Hey ERAA

ERAA
Hi! I'm ERAA! How are you doing?

User
how are you?

ERAA
I'm doing very well. Thanks!

User
open camera

ERAA
opening Camera!!......

User
I want to send a mail

ERAA
opening Gmail....

User
open object detection

ERAA
Launching!!

Send a message

Results

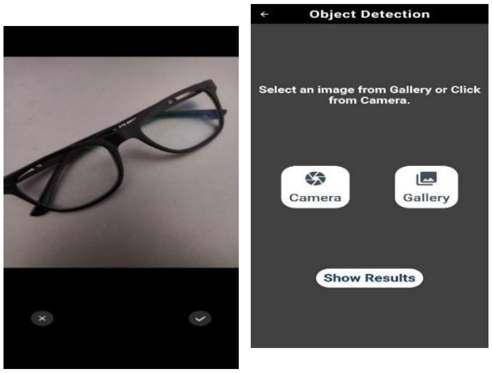
Label: Sunglasses Confidence: 91.41 %

Label: Glasses Confidence: 90.93 %

Label: Goggles Confidence: 88.27 %

Label: Metal Confidence: 62.24 %

Fig – 10(a) Object Detection



Object Detection

Select an image from Gallery or Click from Camera.

Fig – 10(b) Object Detection

After selecting an object the application provides the result in a key value form where the key being the label which defines the name of the object and the value being the confidence score of each label which helps the user in detecting the object.

V. Conclusion

In our Project the application, ERAA, developed with the help of Google Dialogflow was able to perform various tasks like accessing the other applications like WhatsApp, Instagram, Gmail that are installed in the device. Its User-friendly Platform developed with the help of Flutter provided ease in accessing the Application. With the help of Graphic packages in Flutter we were able to provide an attractive user-interface. It was able to perform the basic features as required in an ideal Personal Assistant. The Speech Recognition feature in the application allowed the user to perform the tasks by giving Voice Commands. The Application was also capable of handling small talk with the user. With the development of the Application, we were able to gain enough knowledge on Natural Language Understanding Platform and Machine Learning Models which are the foundations for developing future Artificial Intelligence Models.

References

[1] Mohit Bansal, Dr. T. K. Thivakaran, "Analysis of Speech Recognition using Convolutional Neural Network", Journal of Engineering Sciences, Vol 11, Issue 1, 2020, Page 285-291.

[2] J. Huang, J. Li and Y. Gong, "An analysis of convolutional neural networks for speech recognition," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, South Brisbane, QLD, Australia, 2015, pp 4989-4993, doi: 10.1109/ICASSP.2015.7178920

[3] T. B. Mokgonyane, T. J. Sefara, T. I. Modipa, M. M. Mogale, M. J. Manamela and P. J. Manamela, "Automatic Speaker Recognition System based on Machine Learning Algorithms," *2019 Southern African Universities Power Engineering Conference/Robotics and Mechatronics/Pattern Recognition Association of South Africa (SAUPEC/RobMech/PRASA)*, Bloemfontein, South Africa, 2019, pp. 141-146, doi: 10.1109/RoboMech.2019.8704837.

[4] Vineel Pratap, Qiantong Xu, Jacob Kahn, Gilad Avidov, Tatiana Likhomaneko, Awni Hannun, Vitaliy Liptchinsky, Gabriel Synnaeve, Ronan Collobert, " Scaling up Online Speech Recognition Systems using ConvNets", 27th January 2020.

[5] A. B. Nassif, I. Shahin, I. Attili, M. Azzeh and K. Shaalan, "Speech Recognition Using Deep Neural Networks: A Systematic Review," in *IEEE Access*, vol. 7, pp. 19143-19165, 2019, doi: 10.1109/ACCESS.2019.2896880.

[6] M. A. Khan, A. Tripathi, A. Dixit and M. Dixit, "Correlative Analysis and Impact of Intelligent Virtual Assistants on Machine Learning," *2019 11th International Conference on Computational Intelligence and Communication Networks (CICN)*, Honolulu, HI, USA, 2019, pp. 133-139, doi: 10.1109/CICN.2019.8902424.

[7] Tulshan A.S., Dhage S.N. (2019) Survey on Virtual Assistant: Google Assistant, Siri, Cortana, Alexa. In: Thampi S., Marques O., Krishnan S., Li KC., Ciuonzo D., Kolekar M. (eds) Advances in Signal Processing and Intelligent Recognition Systems. SIRS 2018. Communications in Computer and Information Science, vol 968. Springer, Singapore.

[8] N. A. Godse, S. Deodhar, S. Raut and P. Jagdale, "Implementation of Chatbot for ITSM Application Using IBM Watson," *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, Pune, India, 2018, pp. 1-5, doi: 10.1109/ICCUBEA.2018.8697411.

[9] Linda W. Lee, Amir Dabirian, Iran Paul McCarthy, Jan Kietzmann. (2020), " Making sense of text: artificial intelligence-enabled content analysis", European Journal of Marketing, Vol.54 No.3, pp 615-644.

[10] Roberto Reyes, David Garza, Leonardo Garrido, Victor De la Cueva and Jorge Ramirez, "Methodology for the Implementation of Virtual Assistants for Education Using Google Dialogflow.", Advances in Soft Computing (pp.440-451).

[11] Chinnapa Reddy Kanakanti and Sabitha R., "Ai and Ml Based Google Assistant for an Organization using Google Cloud Platform and Dialogflow", International Journal of Recent Technology and Engineering (IJRTE), Volume-8 Issue-5, January 2020, Page 2722-2727

[12] Mayank Aggarwal and Mani Madhukar, "IBM's Watson Analytics for Health Care: A Miracle Made True.", Cloud Computing Systems and Applications in Healthcare. DOI: 10.4018/978-1-5225-1002-4.ch007.

[13] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Contextdependent pre-trained deep neural networks for largevocabulary speech recognition," IEEE Trans. on Audio, Speech and Language Processing, vol. 20, no. 1, pp. 30– 42, 2012.

[14] Sánchez-Díaz X., Ayala-Bastidas G., Fonseca-Ortiz P., Garrido L. (2018) A Knowledge-Based Methodology for Building a Conversational Chatbot as an Intelligent Tutor. In: Batyrshin I., Martínez-Villaseñor M., Ponce Espinosa H. (eds) Advances in Computational Intelligence. MICAI 2018. Lecture Notes in Computer Science, vol 11289. Springer, Cham. https://doi.org/10.1007/978-3-030-04497-8_14.

[15] Winkler, Rainer & Söllner, Matthias. (2018), "Unleashing the Potential of Chatbots in Education: A State-Of-The-Art Analysis", Academy of Management Proceedings. 2018. DOI: 10.5465/AMBPP.2018.15903abstract

[16] A. P. Singh, R. Nath and S. Kumar, "A Survey: Speech Recognition Approaches and Techniques," *2018 5th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, Gorakhpur, India, 2018, pp. 1-4, doi: 10.1109/UPCON.2018.8596954.

[17] Ossama Abdel-Hamid, Abdelrahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, and Dong Yu, "Convolutional Neural Networks for Speech Recognition", IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol.22,2010.

[18] Ying Zhang, Mohammad Pezeshki, Philemon Brakel, Saizheng Zhang, Cesar Laurent Yoshua Bengio, Aaron Courville, "Towards End-to-End Speech Recognition with Deep Convolutional Neural Networks", arXiv:1701.02720v1,2017.