SMARTBRIDGE INTERNSHIP(AI)

{BATCH 6}

**PROJECT REPORT**

**TOPIC: Customer Help Desk Bot using AI**

Submitted by,

**HASNA PARVEEN KZ**

EMAIL: Hasnaparveen1999@gmail.com

Batch 6(AI)

Smartinterns ID: [SI05202000290@smartinternz.com](mailto:SI05202000290@smartinternz.com)

**CONTENT**

1. INTRODUCTION

1.1 Overview

1.2 Purpose

2. LITERATURE SURVEY

2.1 Existing problem

2.2 Proposed solution

3. THEORITICAL ANALYSIS

3.1 Block diagram

3.2 Hardware / Software designing

4. FLOWCHART

5. RESULT

6. ADVANTAGES & DISADVANTAGES

7. APPLICATIONS

8. CONCLUSION

9. FUTURE SCOPE

10. BIBILOGRAPHY

APPENDIX

A. Source code

1. INTRODUCTION:

1.1OVERVIEW:

Many everyday influences of artificial intelligence are altering the way our daily lives look. If someone from the 1950s travelled through time and arrived in 2019, they would marvel at the way we use our smartphones to navigate around town, how virtual digital assistants such as Alexa and Cortana respond to our queries and would be baffled by our addiction to social media channels such as Facebook, Instagram and Twitter. What is now normal to us and powered by AI, would be utterly foreign to our friend from the past. There's no doubt that artificial intelligence is an integral part of our daily lives.

 Artificial intelligence is when machines can learn and make decisions similarly to humans. There are many types of artificial intelligence including machine learning, where instead of being programmed what to think, machines can observe, analyse and learn from data and mistakes just like our human brains can. This technology is influencing consumer products and has led to significant breakthroughs in healthcare and physics as well as altered industries as diverse as manufacturing, finance and retail. In part due to the tremendous amount of data we generate every day and the computing power available, artificial intelligence has exploded in recent years. We might still be years away from generalised AI—when a machine can do anything a human brain can do—, but AI in its current form is still an essential part of our world.

* 1. PURPOSE:

The typical customer care chatbot can answer simple questions, such as store locations and hours, directions, and maybe even making appointments. When a question falls outside of the scope of the pre-determined question set, the option is typically to tell the customer the question isn’t valid or offer to speak to a real person.

In this project, there will be another option. If the customer question is about the operation of a device, the application shall pass the question onto Watson Discovery Service, which has been pre-loaded with the device’s owners manual. So now, instead of “Would you like to speak to a customer representative?” we can return relevant sections of the owners manual to help solve our customers’ problems.

To take it a step further, the project shall use the Smart Document Understanding feature of Watson Discovery to train it on what text in the owners manual is important and what is not. This will improve the answers returned from the queries.

2. LITERATURE SURVEY

2.1 EXISTING PROBLEM:

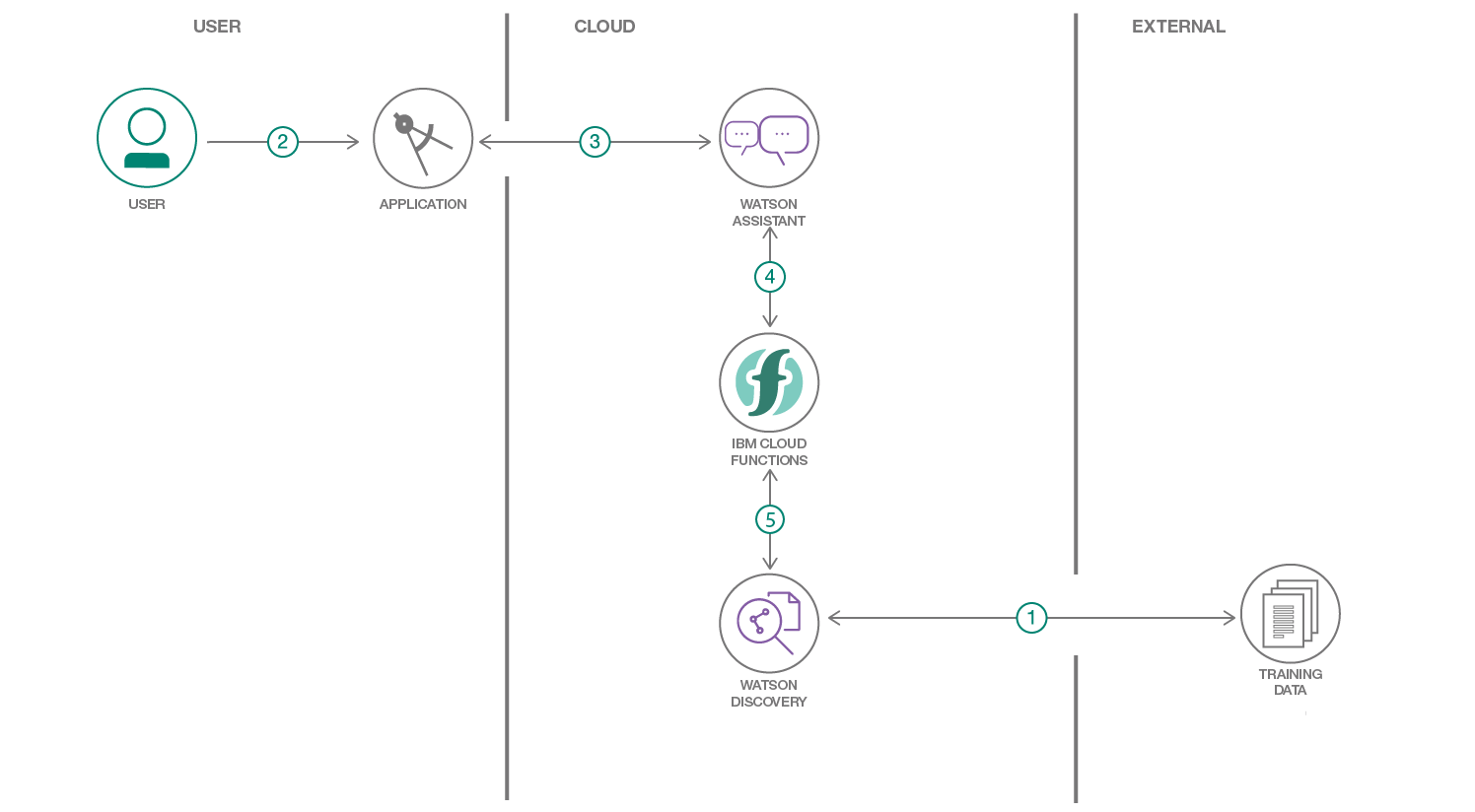
Artificial Intelligence is currently being deployed in customer service to both augment and replace human agents – with the primary goals of improving the customer experience and reducing human customer service costs. While the technology is not yet able to perform all the tasks a human customer service representative could, many consumer requests are very simple ask that sometimes be handled by current AI technologies without human input. Therefore, we need to develop a chatbot that can attend any queries regarding the product including obsolete questions.

2.2 PROPOSED SOLUTION

* Create a customer care dialog skill in Watson Assistant
* Use Smart Document Understanding to build an enhanced Watson Discovery collection
* Create an IBM Cloud Functions web action that allows Watson Assistant to post queries to Watson Discovery
* Build a web application with integration to all these services & deploy the same on IBM Cloud Platform

3. THEORETICAL ANALYSIS

3.1 BLOCK DIAGRAM



* Create a customer care dialog skill in Watson Assistant
* Use Smart Document Understanding to build an enhanced Watson Discovery collection
* Create an IBM Cloud Functions web action that allows Watson Assistant to post queries to Watson Discovery
* Build a web application with integration to all these services & deploy the same on IBM Cloud Platform

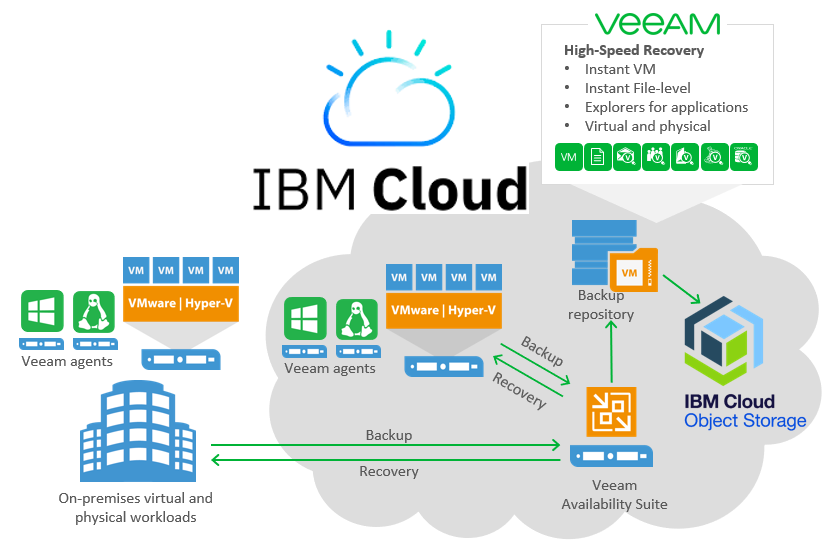
3.2 HARDWARE/SOFTWARE DESIGNING:

A) IBM CLOUD:

The IBM® cloud platform combines platform as a service (PaaS) with infrastructure as a service (IaaS) to provide an integrated experience. The platform scales and supports both small development teams and organizations, and large enterprise businesses. Globally deployed across data centers around the world, the solution you build on IBM Cloud™ spins up fast and performs reliably in a tested and supported environment you can trust.

The platform is built to support your needs whether it's working only in the public cloud or taking advantage of a multicloud deployment model. With our open-source technologies, such as Kubernetes, Red Hat OpenShift, and a full range of compute options, including virtual machines, containers, bare metal, and serverless, you have as much control and flexibility as you need to support workloads in your hybrid environment. You can deploy cloud-native apps while also ensuring workload portability.

Whether you need to migrate apps to the cloud, modernize your existing apps by using cloud services, ensure data resiliency against regional failure, or leverage new paradigms and deployment topologies to innovate and build your cloud-native apps, the platform's open architecture is built to accommodate your use case.

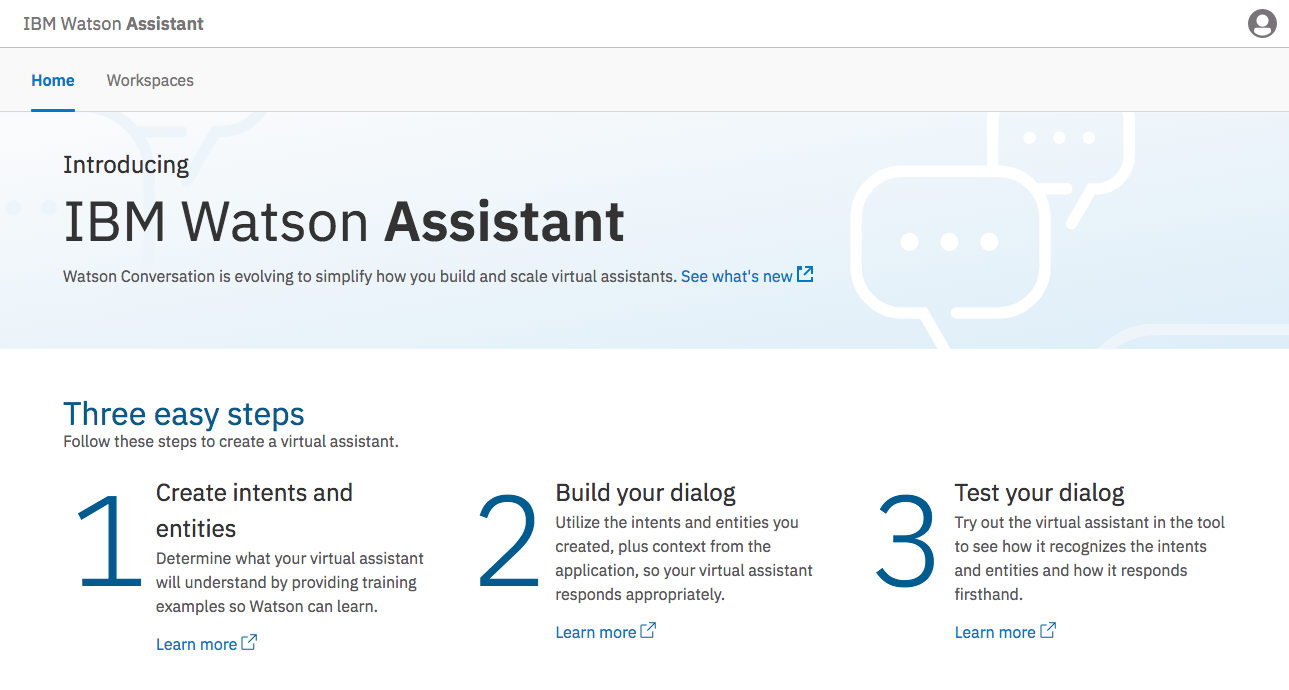


B) IBM WATSON ASSISTANT

Watson Assistant is a conversation AI platform that helps you provide customers fast, straightforward and accurate answers to their questions, across any application, device or channel. By addressing common customer inquiries, Watson Assistant reduces the cost of customer interactions, helping your agents focus on complex use cases – not repetitive responses.

Most chatbots try to mimic human interactions, frustrating customers when a misunderstanding arises. Watson Assistant is more than a chatbot. It knows when to search for an answer from a knowledge base, when to ask for clarity and when to direct users to a human. And since it can be deployed in any cloud or on-premises environment – smarter AI is finally available wherever you need it

* Users interact with the assistant through one or more of these **integration** points:
  + A virtual assistant that you publish directly to an existing social media messaging platform, such as Slack or Facebook Messenger.
  + A custom application that you develop, such as a mobile app or a robot with a voice interface.
* The **assistant** receives user input and routes it to the dialog skill.
* The **dialog skill** interprets the user input further, then directs the flow of the conversation. The dialog gathers any information it needs to respond or perform a transaction on the user's behalf.
* Any questions that cannot be answered by the dialog skill are sent to the **search skill**, which finds relevant answers by searching the company knowledge bases that you configure for the purpose.



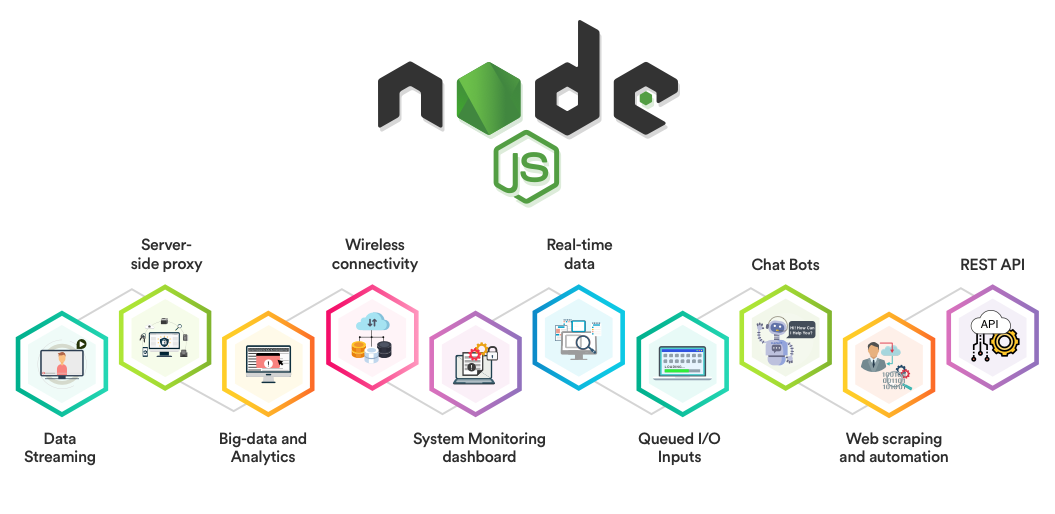
C) NODE.JS

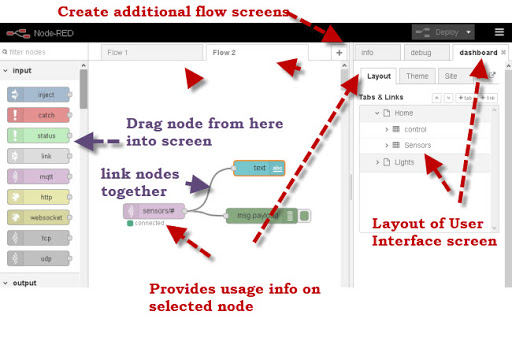
**Node.js** is an [open-source](https://en.wikipedia.org/wiki/Open-source_software), [cross-platform](https://en.wikipedia.org/wiki/Cross-platform), [JavaScript](https://en.wikipedia.org/wiki/JavaScript) runtime environment that executes JavaScript code outside a [web browser](https://en.wikipedia.org/wiki/Web_browser). Node.js lets developers use JavaScript to write command line tools and for [server-side scripting](https://en.wikipedia.org/wiki/Server-side_scripting)—running scripts server-side to produce [dynamic web page](https://en.wikipedia.org/wiki/Dynamic_web_page) content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm,[[6]](https://en.wikipedia.org/wiki/Node.js" \l "cite_note-6) unifying [web-application](https://en.wikipedia.org/wiki/Web_application) development around a single programming language, rather than different languages for server- and client-side scripts.

Though .js is the standard [filename extension](https://en.wikipedia.org/wiki/Filename_extension) for JavaScript code, the name "Node.js" doesn't refer to a particular file in this context and is merely the name of the product. Node.js has an [event-driven architecture](https://en.wikipedia.org/wiki/Event-driven_architecture) capable of [asynchronous I/O](https://en.wikipedia.org/wiki/Asynchronous_I/O). These design choices aim to optimize [throughput](https://en.wikipedia.org/wiki/Throughput) and [scalability](https://en.wikipedia.org/wiki/Scalability) in web applications with many input/output operations, as well as for [real-time Web](https://en.wikipedia.org/wiki/Real-time_Web) applications (e.g., [real-time communication](https://en.wikipedia.org/wiki/Real-time_communication) programs and [browser games](https://en.wikipedia.org/wiki/Browser_game)).[[7]](https://en.wikipedia.org/wiki/Node.js#cite_note-readwrite-7)

The Node.js [distributed development](https://en.wikipedia.org/wiki/Distributed_development) project was previously governed by the Node.js Foundation,[[8]](https://en.wikipedia.org/wiki/Node.js#cite_note-8) and has now merged with the [JS Foundation](https://en.wikipedia.org/wiki/JS_Foundation) to form the [OpenJS Foundation](https://en.wikipedia.org/wiki/OpenJS_Foundation" \o "OpenJS Foundation), which is facilitated by the [Linux Foundation](https://en.wikipedia.org/wiki/Linux_Foundation)'s Collaborative Projects program.[[9]](https://en.wikipedia.org/wiki/Node.js#cite_note-9)

Corporate users of Node.js software include [GoDaddy](https://en.wikipedia.org/wiki/GoDaddy),[[10]](https://en.wikipedia.org/wiki/Node.js#cite_note-10) [Groupon](https://en.wikipedia.org/wiki/Groupon),[[11]](https://en.wikipedia.org/wiki/Node.js#cite_note-11) [IBM](https://en.wikipedia.org/wiki/IBM),[[12]](https://en.wikipedia.org/wiki/Node.js#cite_note-12) [LinkedIn](https://en.wikipedia.org/wiki/LinkedIn),[[13]](https://en.wikipedia.org/wiki/Node.js#cite_note-vbliipad-13)[[14]](https://en.wikipedia.org/wiki/Node.js#cite_note-14) [Microsoft](https://en.wikipedia.org/wiki/Microsoft),[[15]](https://en.wikipedia.org/wiki/Node.js#cite_note-guardianMS-15)[[16]](https://en.wikipedia.org/wiki/Node.js#cite_note-16) [Netflix](https://en.wikipedia.org/wiki/Netflix),[[17]](https://en.wikipedia.org/wiki/Node.js#cite_note-17) [PayPal](https://en.wikipedia.org/wiki/PayPal),[[18]](https://en.wikipedia.org/wiki/Node.js#cite_note-18)[[19]](https://en.wikipedia.org/wiki/Node.js#cite_note-19) [Rakuten](https://en.wikipedia.org/wiki/Rakuten), [SAP](https://en.wikipedia.org/wiki/SAP_SE),[[20]](https://en.wikipedia.org/wiki/Node.js#cite_note-sapappbuilder-20) [Voxer](https://en.wikipedia.org/wiki/Voxer),[[21]](https://en.wikipedia.org/wiki/Node.js#cite_note-21) [Walmart](https://en.wikipedia.org/wiki/Walmart),[[22]](https://en.wikipedia.org/wiki/Node.js#cite_note-vbwalmart-22) and [Yahoo!](https://en.wikipedia.org/wiki/Yahoo!).[[23]](https://en.wikipedia.org/wiki/Node.js#cite_note-23)



D) DESIGNING NODE RED FLOW: 

**Node-RED** is a [flow-based](https://en.wikipedia.org/wiki/Flow-based_programming) development tool for [visual programming](https://en.wikipedia.org/wiki/Visual_programming_language) developed originally by [IBM](https://en.wikipedia.org/wiki/IBM) for wiring together hardware devices, [APIs](https://en.wikipedia.org/wiki/Application_programming_interface) and [online services](https://en.wikipedia.org/wiki/Online_services) as part of the [Internet of Things](https://en.wikipedia.org/wiki/Internet_of_Things)

Node-RED provides a [web browser](https://en.wikipedia.org/wiki/Web_browser)-based flow editor, which can be used to create [JavaScript](https://en.wikipedia.org/wiki/JavaScript) functions. Elements of applications can be saved or shared for re-use. The runtime is built on [Node.js](https://en.wikipedia.org/wiki/Node.js). The flows created in Node-RED are stored using [JSON](https://en.wikipedia.org/wiki/JSON). Since version 0.14, [MQTT](https://en.wikipedia.org/wiki/MQTT) nodes can make properly configured [TLS](https://en.wikipedia.org/wiki/Transport_Layer_Security) connections.

5. FLOWCHART

6) ADVANTAGES AND DISADVANTAGES:

**1) Reduction in Human Error:**

The phrase “**human error**” was born because humans make mistakes from time to time. Computers, however, do not make these mistakes if they are programmed properly. With Artificial intelligence, the decisions are taken from the previously gathered information applying a certain set of algorithms. So errors are reduced and the chance of reaching accuracy with a greater degree of precision is a possibility.

**2) Takes risks instead of Humans:**

This is one of the biggest advantages of Artificial intelligence. We can overcome many risky limitations of humans by developing an AI Robot which in turn can do the risky things for us. Let it be going to mars, defuse a bomb, explore the deepest parts of oceans, mining for coal and oil, it can be used effectively in any kind of natural or man-made disasters.

**3) Available 24x7:**

An Average human will work for 4–6 hours a day excluding the breaks. Humans are built in such a way to get some time out for refreshing themselves and get ready for a new day of work and they even have weekly offed to stay intact with their work-life and personal life. But using AI we can make machines work 24x7 without any breaks and they don’t even get bored, unlike humans.

**4) Helping in Repetitive Jobs:**

In our day-to-day work, we will be performing many repetitive works like sending a thanking mail, verifying certain documents for errors and many more things. Using artificial intelligence we can productively automate these mundane tasks and can even remove “**boring**” tasks for humans and free them up to be increasingly creative.

**5) Digital Assistance:**

Some of the highly advanced organizations use digital assistants to interact with users which saves the need for human resources. The digital assistants also used in many websites to provide things that users want. We can chat with them about what we are looking for. Some chatbots are designed in such a way that it’s become hard to determine that we’re chatting with a chatbot or a human being.

**6) Faster Decisions:**

Using AI alongside other technologies we can make machines take decisions faster than a human and carry out actions quicker. While taking a decision human will analyze many factors both emotionally and practically but AI-powered machine works on what it is programmed and delivers the results in a faster way.

**7) Daily Applications:**

Daily applications such as Apple’s **Siri**, Window’s **Cortana**, Google’s **OK Google** are frequently used in our daily routine whether it is for searching a location, taking a selfie, making a phone call, replying to a mail and many more.

**Example:**Around 20 years ago, when we are planning to go somewhere we used to ask a person who already went there for the directions. But now all we have to do is say “**OK Google**where is Visakhapatnam”. It will show you Visakhapatnam’s location on google map and the best path between you and Visakhapatnam.

**8) New Inventions:**

AI is powering many inventions in almost every domain which will help humans solve the majority of complex problems.

**Example:**Recently doctors can predict breast cancer in the woman at earlier stages using advanced AI-based technologies.

As every bright side has a darker version in it. Artificial Intelligence also has some **disadvantages.** Let’s see some of them

**1) High Costs of Creation:**

As AI is updating every day the hardware and software need to get updated with time to meet the latest requirements. Machines need repairing and maintenance which need plenty of costs. It’ s creation requires huge costs as they are very complex machines.

**2) Making Humans Lazy:**

AI is making humans lazy with its applications automating the majority of the work. Humans tend to get **addicted**to these inventions which can cause a problem to future generations.

**3) Unemployment:**

As AI is replacing the majority of the repetitive tasks and other works with robots,human interference is becoming less which will cause a major problem in the employment standards. Every organization is looking to replace the minimum qualified individuals with AI robots which can do similar work with more efficiency.

**4) No Emotions:**

There is no doubt that machines are much better when it comes to working efficiently but they cannot replace the human connection that makes the team. Machines cannot develop a bond with humans which is an essential attribute when comes to Team Management.

**5) Lacking Out of Box Thinking:**

Machines can perform only those tasks which they are designed or programmed to do, anything out of that they tend to crash or give irrelevant outputs which could be a major backdrop.

7. APPLICATIONS

-It becomes so easy to get the details on anything based on the user manual for customer care helpdesk.

Various functions like configurations of wifi, setup steps, accessing settings, configuration information etc can be addressed so easily.

-with the advent of a customer care chatbot, user can access it 24X7 .

-Not only for customer care services, this technology finds a wide application in many fields like temperature and atmosphere predictions, translation of a language into any language and many more….

8. CONCLUSION:

Artificial Intelligence and Machine Learning are products of both science and myth. The idea that machines could think and perform tasks just as humans do is thousands of years old. The cognitive truths expressed in AI and Machine Learning systems are not new either. It may be better to view these technologies as the implementation of powerful and long-established cognitive principles through engineering.

9. FUTURE SCOPE:

From gauging purchase intent to answering questions about IT issues, chatbots are on track to play a major role in the contemporary enterprise. Chatbots are fully functioning, semi-autonomous systems that can assist customer service experiences and response time.

But that doesn't mean their future in the enterprise is secure. For [chatbots](https://searchcustomerexperience.techtarget.com/definition/chatbot) to withstand the rapidly increasing technological shifts and become mainstays in the enterprise, developers need to examine the issues that have popped up with increased implementation.

The future scope of chatbots could include many benefits for enterprises, but experts say they will need to be gently nudged in the right direction for businesses to reap these benefits.

The **future of chatbots** is that businesses will automate simple payments and allow users to pay directly over live chat or Facebook Messenger apps. The instant process makes the customer happy and improves customer satisfaction.

## **Chatbots Now and In Our Future**

Business applications of chatbots for consumer-facing goods are growing rapidly.

In fact, over 59% of millennials and 60% of Gen Xers in the United States have [interacted with chatbots](http://www.businessinsider.com/chatbot-monetization-market-business-strategies-opportunites-2017-11).

And according to a [Facebook survey,](https://messenger.fb.com/) more than 50% of customers say they’re more likely to shop with a business that they can connect with via chat.

According to [Gartner](https://www.gartner.com/imagesrv/summits/docs/na/customer-360/C360_2011_brochure_FINAL.pdf#page=2),

*“By 2020, 85% of our engagement with businesses will be done without interacting with another human. Instead, we’ll be using self-service options and chatbots.”*

Customers expect to be able to find the information they’re looking for in a click of a button and in the blink of an eye.

When this isn’t possible, frustration brews, and this can lead to you losing a sale or even [losing a potential customer forever](https://www.bigcommerce.com/ecommerce-answers/what-customer-attrition/).

To understand how chatbots can mitigate some of these frustrations and improve the user experience, we need to first look at how customers are choosing to interact with businesses today.

In the 2017 Global State of Customer Service Report by Microsoft, we can see how email and telephone are still the dominant communication channels for many customers.

And while live chat, self-service, social media, and chatbots are relatively lower in terms of raw volume, the growth of these channels are staggering

10. BIBILOGRAPHY:

<https://github.com/SmartPracticeschool/llSPS-INT-1779-Intelligent-Customer-Help-Desk-with-Smart-Document-Understanding/issues>

<https://cloud.ibm.com/docs/overview?topic=overview-whatis-platform>

<https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application/>

<https://www.w3schools.com/howto/howto_make_a_website.asp>

<https://www.ibm.com/watson/products-services>

<https://developer.ibm.com/technologies/web-development/articles/ws-restful>

APPENDIX:

SOURCE CODE:

|  |
| --- |
| /\*\* |
|  | \* |
|  | \* @param {object} params |
|  | \* @param {string} params.iam\_apikey |
|  | \* @param {string} params.url |
|  | \* @param {string} params.username |
|  | \* @param {string} params.password |
|  | \* @param {string} params.environment\_id |
|  | \* @param {string} params.collection\_id |
|  | \* @param {string} params.configuration\_id |
|  | \* @param {string} params.input |
|  | \* |
|  | \* @return {object} |
|  | \* |
|  | \*/ |
|  |  |
|  | const assert = require('assert'); |
|  | const DiscoveryV1 = require('watson-developer-cloud/discovery/v1'); |
|  |  |
|  | /\*\* |
|  | \* |
|  | \* main() will be run when you invoke this action |
|  | \* |
|  | \* @param Cloud Functions actions accept a single parameter, which must be a JSON object. |
|  | \* |
|  | \* @return The output of this action, which must be a JSON object. |
|  | \* |
|  | \*/ |
|  | function main(params) { |
|  | return new Promise(function (resolve, reject) { |
|  |  |
|  | let discovery; |
|  |  |
|  | if (params.iam\_apikey){ |
|  | discovery = new DiscoveryV1({ |
|  | 'iam\_apikey': params.iam\_apikey, |
|  | 'url': params.url, |
|  | 'version': '2019-03-25' |
|  | }); |
|  | } |
|  | else { |
|  | discovery = new DiscoveryV1({ |
|  | 'username': params.username, |
|  | 'password': params.password, |
|  | 'url': params.url, |
|  | 'version': '2019-03-25' |
|  | }); |
|  | } |
|  |  |
|  | discovery.query({ |
|  | 'environment\_id': params.environment\_id, |
|  | 'collection\_id': params.collection\_id, |
|  | 'natural\_language\_query': params.input, |
|  | 'passages': true, |
|  | 'count': 3, |
|  | 'passages\_count': 3 |
|  | }, function(err, data) { |
|  | if (err) { |
|  | return reject(err); |
|  | } |
|  | return resolve(data); |
|  | }); |
|  | }); |
|  | } |