



CHATBOT (I) FOR GOOGLE KUBERNETES

Kashish Deshwal¹, Mehul Varshney², Paras Lakhani³, Deepika Gupta⁴

^{1,2,3,4}CSE Department, MIET, Meerut, (India)

ABSTRACT

Chatbot are artificial intelligence (AI) software, they instigate conversation with the users in natural language with the help of websites, mobile apps and messaging applications. Chatbot furnish companies new opportunities to improve the customers engagement process and operational efficiency by reducing the typical cost of customer service. Here, we present the delineation and development of chatbots, there uses and the spheres where chatbots are used. Then elaborating the use of NLP (Natural Language Processing) in fabrication of chatbots. Our system will work for google Kubernetes, as there isn't any chatbot and all the work in Kubernetes has to done manually. This will be used for industry purpose.

KEYWORDS: AI, Chatbot, Google Kubernetes, Natural Language Processing.

1.INTRODUCTION

Chatbots are widely used to perform an automated task chatbot is merely a computer program that fundamentally simulates human conversations. It facilitates communication between a human and a machine. The communication can happen via messages or voice command. A chatbot is programmed to work independently from a human operator so as to save time of humans. These days, we see the chat bots are ubiquitous. Chat bots answers all the users questions for each and every domain where it is operating. Chatbots commune in a specific way using natural language. Chatbots are digital solution to improve the information acquisition.

The basic technologies that we are going to work on our project are Python, machine learning and Kubernetes. Let's see the brief introduction of all of them-

1.1 PYTHON

Python is a language which helps us builds the conversational components. Web server is a platform to host components. Modules and packages that Python supports persuades the re-usage and modularity of any program. The most important aspect of chatbots, i.e., Natural language processing (NLP) toolkit, is written in Python.



1.2 MACHINE LEARNING

Machine learning is widely used in chatbots so as to response in effective manner as ML has the quality of learning and improving instinctively. It is not programmed specifically.

1.3 KUBERNETES

Kubernetes uses GOOGLE infrastructure to provides a platform for effective managing, scaling and deploying the applications which are containerize applications. The container cluster is formed by the help of different machines which exists in Kubernetes engine environment.

2. RELATED WORK

Digging deep into the history of chatbots, Alan Turing's printed an editorial "Computing Machinery and Intelligence" in year 1950. This paper tells about fundamentals of intelligence that has taken into thought of the Turing test. This basis was dependent on period of time and voice communication of laptop programs to imitate like humans and a personality's choose, adequately it had been not possible to differentiate by human choose alone on the idea of voice communication that it had been between the program and a true human.

ELIZA program came up in 1966 by Joseph Weizenbaum thanks to his keen interest in Turing's take a look at. This program was quite asuccess in convincing that the voice communication is between true human solely and not a larva.

However, once a specific program is unmasked, once its inner workings area unit explained and its magic crumbles away. The observer said "I may have written that". Therewith thought he moves the program in question from the shelf marked "intelligent", thereto reserve for curios, the item of this paper is to cause simply such a re-evaluation of the program near to be "explained". Few programs ever required it a lot of.

ELIZA's key technique of operation involves the popularity of clue words or phrases within the input, and also the output of corresponding pre-prepared or pre-programmed responses which will move the voice communication forward in Associate in Nursing apparently purposeful means (e.g. by responding to any input that contains the word 'MOTHER' with 'TELL Pine Tree State a lot of regarding YOUR FAMILY'). Therefore, Associate in Nursing illusion of understanding is generated, despite the fact that the process concerned has been simply superficial. ELIZA showed that such Associate in Nursing illusion is astonishingly simple to come up with, as a result of human judges area unit therefore able to offer the advantage of the doubt once colloquial responses area unit capable of being taken as "intelligent".

Interface designers have return to understand that humans' readiness to interpret laptop output as genuinely conversational—even once it's really supported rather straightforward pattern-matching—can be exploited for helpful functions. the majority value more highly to interact with programs that area unit human-like, and this



provides chatbot-style techniques a doubtless helpful role in interactive systems that require to elicit data from users, as long as that data is comparatively easy and falls into sure classes. Thus, for instance, on-line facilitate systems will usefully use chatbot techniques to spot the realm of facilitate that users need, doubtless providing a "friendlier" interface than a lot of formal search or menu system. this kind of usage holds the prospect of moving chatbot technology from Weizenbaum's "shelf reserved for curios" thereto marked "genuinely helpful process methods".[1]

3. VARIANTS OF CHATBOT

To know about the various modules used in chatbot, let's talk about two variants, i.e., rule-based and self-learning.

In rule-based approach, a bot is trained and answerable on the basis of rules. These rules vary from simple to complex. Although the bots handle simple queries but fail to manage complex ones. On the other hand, self-learning bots use Machine Learning-based approaches which are for sure more efficient than rule-based bots. These bots can be of further two types: **Retrieval Based** or **Generative based**,

This paper proposed the retrieval based chatbot which will be self-learning bot and modules include- NLTK, Scikit library, NumPy, SpaCy, Client, Config, Kubectl, Kubeadm

Each one of them can be explained as-

3.1 NLP

Neuro-Linguistic Programming (NLP) where Neuro defines neurology, Linguistic is referred to language and programming defines how the neural language works. NLP is the most prominent way which comes into use when anyone can't understand language. It is the way one communicates with anyone and is developed after getting proper outcome from clients by extraordinary communicators and therapists. NLP is a kind of methodology which tells us the way to accomplish goals and get outcome.

3.2 NLTK

Natural Language Toolkit (NLTK) is one of the most substantial NLP libraries that consists of packages like Tokenization, Stemming, Lemmatization, Punctuation, Character count, word count help the machines to acknowledge human language and response to it correctly.

3.3 SCIKIT LIBRARY

It is a library which is helpful in providing unsupervised and supervised learning algorithm. It's based on technologies NumPy, pandas, and Matplotlib. Some functionality of this library are-

Regression, Classification, Clustering and model selection preprocessing.

3.4 NUMPY

NumPy provides multidimensional array object of high-performance and is an array-processing package. It is used for scientific computing. Some of the most important features are-

3.4.1 N-dimensional array object



3.4.2. Sophisticated functions

3.4.3 Tools for integrating C/C++ and Fortran code

3.4.4 It has useful linear algebra

3.5 SPACY

It is extremely accurate library used in deep learning

3.6 CLIENT

Kubernetes API client is generated by KUBERNETES -CLIENT only at run-time and that too on the basis of specifications of OpenAPI. We can achieve the task of creating a client from cluster's API specification and cluster's kubeconfig file.

For using the Kubernetes API effectively in different programming languages, we have certain client libraries, like-

3.6.1 Officially-supported Kubernetes client libraries.

3.6.2 Community- maintained client libraries

We can use the Kubernetes REST API for writing the applications and also it is not necessary for implementing request/response types and the API calls. It is efficient to use a client library for the programming language that we want to use.

Some very specific and common tasks which we want to perform like authentication can be done by Client libraries. If the client API is running inside the Kubernetes cluster, the client libraries can be used for authenticating and understanding the kubeconfig and it uses the Kubernetes service account.

3.7 CONFIG

We use the configparser module to work with and on configuration files. The configuration files are arranged into sections, and each section can carry name-value pairs which are used for configuration data. **Config** file sections can be recognised by looking for lines which starts with [and ends with].

GENERAL CONFIGURATION CLUES

3.7.1 We need to state the latest stable version of API, during the defining config.

3.7.2 YAML must be used for writing configs in spite of using JSON.

3.7.3 There is no need to describe the default values unnecessarily.

3.7.4 Before pushing the config file to the cluster, it is important to store these config files in version control.

3.8 KUBECTL

Kubectl which is widely used for running commands in opposition to Kubernetes clusters is a command line interface. Kubectl searches in the \$HOME/kube directory for a file named config.

3.8.1 Syntax for kubectl commands are as follows-

Kubectl [command] [TYPE] [NAME] [flags]

Where we can describe each of as follows-



Command- It specifies exactly what operation we want to carry out on the resources. Example- get, describe

TYPE- It specifies what resource type we are using. These are case- sensitive and we are free to state the plural, singular and abbreviated forms. Example- If we have the commands as follow, we will see the same output-

Kubectl get pod pod1

Kubectl get pods pod1

NAME- It tells us about the resource name. These names are also case-sensitive. Example- kubectl get pods.

Resources can be stated by name and type as-

1. If we have same type then we can group resources. Example-
Kubectl get pod example-pod1 example-pod2
 2. If we can identify individually multiple resource type. Example-
Kubectl get pod/example-pod1 replicationcontrollee/example-rc1
- Flags- It clarifies optional flags.

OPERATIONS-

Operation	work
-----------	------

1. Kubectl apply	To apply a configuration to resource by filename.
2. Kubectl attach	To attach a running container
3. Kubectl cluster-info	To display the cluster information
4. Kubectl config	To modify kubeconfig files
5. Kubectl cp	To copy files and directories to and from containers
6. Kubectlautoscale	to autoscalea deployment, replicaSet
7. Kubectl explain	For documentation of resources
8. Kubectl logs	Print the logs for a container in a pod
9. Kubectl replace	Replace a resource by filename
10.Kubectl convert	Convert config files between different API versions

TABLE

3.9 KUBEADM

Kubeadm is designed for the creation of KUBERNETES CLUSTERS and is most widely used tool in Kubernetes. It is the best-practice “fast paths” so as to provide ‘kubeadm init’ and ‘kubeadm join’. Kubeadm executes the necessary actions so as to have a minimum viable cluster



which is running and up for use. If we see the design considerations then Kubeadm is not at all cautious about provisioning machines but only about bootstrapping. Kubeadm's range does not include several supplements like the Kubernetes dashboard, cloud- specific extras and monitoring solutions and resource management techniques.

4. WORK FLOW OF THE PROPOSED SYSTEM

STEP1- As soon as the person opens the google Kubernetes, chatbot interface appears with a message "hi, how may I help you?"

STEP2- The person can ask for any help related to Kubernetes like creating of clusters, pods, or creating or deleting content etc.

STEP3- In reply the chatbot wants the user to get the access key (if user is registered) and wants the user to sign up first, if unregistered.

STEP4- The chatbot will authenticate the keys entered by the users.

STEP5- Chatbot now performs according to the user.

FOR EXAMPLE- Creating containers, clusters, namespace, pods etc.

4.1 ALGORITHM

The algorithm used for the main component of our system i.e. Kubernetes is as follows-

1. Create function.
2. 'Configs' will set the configuration class.
3. Initialize, `v1=client.CoreV1Api()`, calling kubernetes module client to fetch client ,server status.
4. `ret = v1.list_pod_for_all_namespaces(watch=False)`, finding all the pods with their ip using `list_pod_for_all_namespaces` module
5. Iterate over ret from 3 to 4 to fetch all the data
6. print `pod_ip, namespace, name`

4.2 DATASET

4.2.1 Tag: salutation

Samples: Hiii, Howyoudoing?, What's up?, Hey, Hello

Reaction: Hello, I'm good, Good to visualize you again, Hii, how can I help you?

4.2.2 Tag: gratitude

Samples: Thanking you, Thank you, That's great, Awesome, thanks, Thanks for the help

Reaction: Happy to help!!, Any time!, Always, My pleasure



4.2.3 Tag: Blank

Samples: []

Reaction: Can't/cannotperceive you, Pleaseoffer me more info, What, Not sure, Try again

4.2.4 Tag: cluster_info

Samples: can you tell me something about clusters, what are clusters?,how can clusters be useful to me

Reaction: Cluster contains multiple worker machines which are known as nodes and also minimum of one cluster master. For representation of applications that are containerized by Kubernetes objects falls lead of clusters. GKE is the dissolution of cluster.

4.2.5 Tag: namespace

Samples:what are namespace, can you help me with namespace, why are namespace important in Kubernetes

Reaction: Traversing to namespace module

4.2.6 Tag: pod_ip

Samples: what is the pod status?, give me the pod status please, what's my pod status?

Reaction: Yes, ofcourse, wait for a while

4.2.7 Tag: meta_data_name

Samples: can you help me in receiving meta data of pod, what is meta data name,what is meta data name for my work?

Reaction: yes, I'm ready, I'll tell you

4.2.8 Tag:deployment

Samples:are we ready to deploy, can we start the deployment work, can we do deployment now, is deployment of my work done/completed, what do we need to deploy

Reaction:Let's start the deployment work, Please wait for a few seconds

4.2.9 Tag: create_job

Samples: can you create job for me?,create job, create my job please

Reaction:Yes sure, I'll come soon with the status.

4.2.10 Tag:delete_job

Samples: can you delete the job for me, delete job

Reaction: Yes sure, I'll come soon with the status.

4.2.11 Tag: update_job

Samples: can you update this job, update my job please

Reaction: Yes sure, I'll come soon with the status.

4.2.12 Tag: create_cluster

Samples: I want a cluster to be created for me, create a cluster, create cluster

Reaction: would appreciate your patience, please wait

International Conference on Science, Technology and Management (ICSTM-2020)



Guru Gobind Singh Polytechnic, Nashik, Maharashtra (India)



15th - 16th February 2020

www.conferenceworld.in

ISBN : 978-81-944855-1-3

4.2.13 Tag: farewell

Samples: Byee, departure, See you later, Nice chatting with you

Reaction: See you!!, Have a good day, Bye! Come soon, longing to see you again

5. CONCLUSION

It was impossible for a person involved in Kubernetes to manage their work easily. It was them who has to do all the hectic work from creating a cluster to perform all operations like name-space, pods, its configuration etc. Our chatbot 'I' helps the customers of Kubernetes to manage everything from creating to resource management and monitoring. It is a user-friendly interface to solve all the queries of the customer efficiently.

Summarizing this project, we have learnt a lot about Kubernetes, its main components that helps in forming a cluster like API server, scheduler, controller manager, kubelet, kube-proxy and many more. This project helps us to understand how to integrate Kubernetes with machine learning and uses modules of python and NLP.

This project will benefit a huge number of customers working on Kubernetes and also for the ones who want to work on Kubernetes but find the interface a bit difficult. The user can ask queries like what is Kubernetes or any other queries related to Kubernetes and it helps, performs and after the input all the steps will be done by the chatbot, which is already lined to Kubernetes API.

REFERENCE

[1] <https://en.wikipedia.org/wiki/Chatbot>