

Meetbot : Scheduler Bot integrated with Slack



-Sadakhya Narnur



Agenda - Answering the what and hows!

- What are we building?
- How to build this bot?
- How to integrate the bot with Slack?
- How to deploy on Apprunner?
- How can we improve the bot?

What are we building?

An NLP based bot that helps with scheduling meetings based on its conversation with a user. The bot is essentially a Flask app that is hosted with Ngrok.

An App named 'Meetbot' communicates with our Flask app and responds based on the inference of our model.

How to build this bot?

1. Collect and Prepare the data: Here the intents.json consists of question and answer data that will be used to train the model.
2. Transform the data: The intents are read, word level tokenized and stemmed. We use the Bag-of-words by creating a word dictionary for each document, create one-hot encoded output value and data is pickled.
3. Build DNN model: Once we load the model we fitted the training data and saved the checkpoints and model artifacts.
4. Run inference on user messages and send the response with highest probability.

How to integrate the bot with Slack?

1. Create an app in Slack that is integrated with a workspace and a channel.
2. Add required Bot Scopes to this app like “Send messages as @Meetbot”, “View messages” and “Post messages”.
3. Integrate the Bot OAuth Token, Signing Secret and Webhook url into our Python Flask code.
4. To expose the Flask api to Slack we use ngrok to host our app. This ngrok endpoint is given for Request URL to enable events and subscribe to bot events.
5. Using the slack apis the bot can access the message payloads to reply. The model predicts on the message it received and posts the chat message.

Services | AWS App Runner | Bills | Billing and Cost Manag... | Slack API: Applications | Sai... | Slack - schedule_n_meet - : x +

api.slack.com/apps/A06KSDXHZFV

Slack - schedule_n_meet - : x +

DocumentationTutorialsYour Apps

slack api

Directory

Review & Submit

Give feedback

Stack ♥

Help

Contact

Policies

Our Blog

App Credentials

These credentials allow your app to access the Slack API. They are secret. Please don't share your app credentials with anyone, include them in public code repositories, or store them in insecure ways.

App IDA06KSDXHZFV

Date of App CreationFebruary 20, 2024

Client ID6659138368100.6672473611539

Client Secret

*****ShowRegenerate

You'll need to send this secret along with your client ID when making your `oauth_v2.access` request.

Signing Secret

*****ShowRegenerate

Slack signs the requests we send you using this secret. Confirm that each request comes from Slack by verifying its unique signature.

Verification Token

Tqr5QUMRvkvMOXQsRG8YkWWYRegenerate

This deprecated Verification Token can still be used to verify that requests come from Slack, but we

Discard ChangesSave Changes

Services | AWS App Runner | Bills | Billing and Cost Manag... | Slack API: Applications | Sai... | Slack - schedule_n_meet - : x +

api.slack.com/apps/A06KSDXHZFV/yourApp

Slack - schedule_n_meet - : x +

Save URLL

Scopes

A Slack app's capabilities and permissions are governed by the `scopes` it requests.

Bot Token Scopes

Scopes that govern what your app can access.

OAuth ScopeDescription

channels:historyView messages and other content in public channels that Meetbot has been added to

chat:writeSend messages as @Meetbot

incoming-webhookPost messages to specific channels in Slack

Add an OAuth Scope

User Token Scopes

Scopes that access user data and act on behalf of users that authorize them.

OAuth ScopeDescription

Discard ChangesSave Changes

Services | AWS App Runner | Bills | Billing and Cost Manag... | Slack API: Applications | Sai... | Slack - schedule_n_meet - : x +

api.slack.com/apps/A06KSDXHZFV/yourApp

Slack - schedule_n_meet - : x +

DocumentationTutorialsYour Apps

slack api

Settings

Basic Information

Collaborators

Socket Mode

Install App

Manage Distribution

Features

App Home

Org Level Apps

Incoming Webhooks

Interactivity & Shortcuts

Slash Commands

Workflow Steps

OAuth & Permissions

Event Subscriptions

User ID Translation

App Manifest

Beta Features

Submit to App Directory

Reinstall to Workspace

Review & Submit

Give feedback

Advanced token security via token rotation

Recommended for developers building on or for security-minded organizations - opting into token rotation allows app tokens to automatically expire after they're issued within your app code. View documentation.

At least one redirect URL needs to be set below before this app can be opted into token rotation

Opt In

OAuth Tokens for Your Workspace

These tokens were automatically generated when you installed the app to your team. You can use these to authenticate your app. Learn more.

Bot User OAuth Token

xxxx-6659138368100-6672589994658-llbpCY32UvWtrOVPCwGFAsbCopy

Access Level: Workspace

Reinstall to Workspace

Discard ChangesSave Changes

Services | AWS App Runner | Bills | Billing and Cost Manag... | Slack API: Applications | Sai... | Slack - schedule_n_meet - : x +

api.slack.com/apps/A06KSDXHZFV/event-subscriptions?

Slack - schedule_n_meet - : x +

Enable EventsOn

Basic Information

Collaborators

Socket Mode

Install App

Manage Distribution

Features

App Home

Org Level Apps

Incoming Webhooks

Interactivity & Shortcuts

Slash Commands

Workflow Steps

OAuth & Permissions

Event Subscriptions

User ID Translation

App Manifest

Beta Features

Submit to App Directory

Review & Submit

Give feedback

Stack ♥

Request URL Verified

https://03fb-2601-647-6380-6850-25c0-184c-c799-76a4.ngrok-free.app/sd/Change

We'll send HTTP POST requests to this URL when events occur. As soon as you enter a URL, we'll send a request with a `challenge` parameter, and your endpoint must respond with the challenge value. Learn more.

New event authorization format

Recent changes to Events API payloads

The Events API now sends information about authorized users and workspaces in a new, compact format. Learn more.

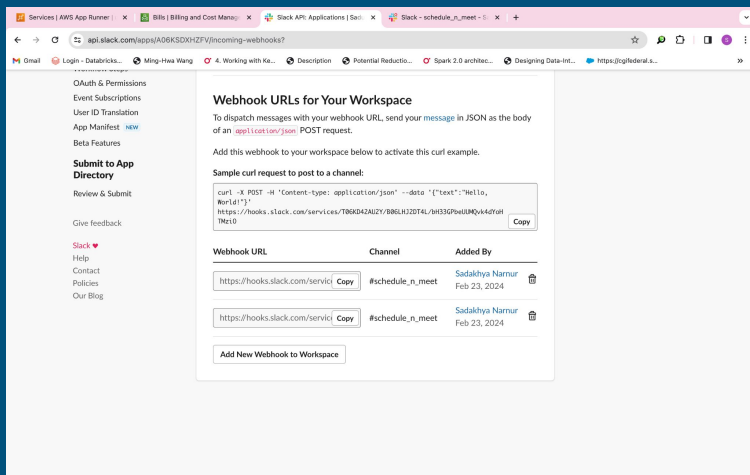
Subscribe to bot events

Apps can subscribe to receive events the bot user has access to (like new messages in a channel). If you add an event here, we'll add the necessary `OAuth scope` for you.

Event NameDescriptionRequired Scope

message:channelsA message was posted to a channelchannels:history

Discard ChangesSave Changes



- Run the flask app on localhost port 5000.
- Run ngrok with:
> ngrok http 5000

How to deploy on Apprunner?

1. Using Github repository: We can create an App runner service with the source code repository pointing to our github repo, set the required environment variables and configurations.
2. Using docker image: The docker image can be pushed to Elastic Container Registry service which can be provided as Container Registry to the App runner service. Further setting the configurations we can build.

Provide the public endpoint to the Slack event subscription request URL. Thereby using the public endpoint the app can communicate with slack.

How to improve the bot?

- Data : The larger the data the better the model would learn, hence the model needs more training data in intents file. The bot needs variety of data thereby improving quality of training.
- Model : The current implementation is a DNN. In order to improve the bot performance other QA NLP models like BERT can be considered.



Demo Time!