PROJECT REPORT

PRACTICE MODULE FOR CERTIFICATE IN:

INTELLIGENT SOFTWARE AGENTS (ISA)

TaskSlinger

Task management made simple for people on the go!

Project Report

Team Members: Richard Chai

Contents

1.0 EXECUTIVE SUMMARY

TaskSlinger is an intelligent chatbot that takes voice and text input to help users create tasks in their Google Calendar.

This can benefit a wide range of individuals, including:

Busy professionals:

- They often have packed schedules and numerous commitments. TaskSlinger can quickly and conveniently add tasks to their calendar without having to manually type them out, saving time and ensuring they don't miss any important appointments.

Individuals with disabilities:

- Those with visual or motor impairments may find it challenging to use a traditional calendar interface. A voice-enabled chatbot provides an accessible alternative, allowing them to manage their calendar hands-free.

Elderly individuals:

- Older adults may find technology challenging to navigate. A simple voice-based interface for adding calendar events can be more intuitive and user-friendly for this demographic.

Multitaskers:

- People who are often on the go or juggling multiple tasks can benefit from the convenience of adding calendar events via voice or text. They can quickly create reminders or schedule appointments without interrupting their current activity.

Students:

- Students often have busy schedules with classes, assignments, and extracurricular activities. A chatbot can help them stay organized by quickly adding deadlines, exam dates, or social events to their calendar.

An intelligent chatbot like TaskSlinger that takes voice and text input to create tasks in Google Calendar can provide a more efficient, accessible, and user-friendly experience for individuals with busy schedules, disabilities, or those who simply prefer a more conversational way of interacting with technology.

2.0 PROBLEM STATEMENT / BUSINESS OPPORTUNITY

People today are living increasingly busy lives, in crowded cities, spending a lot of time commuting to and from work in transportation that may not be conducive to easy manipulation of digital devices.

Benefits of TaskSlinger include:

Efficiency:

- Users can save time by quickly creating calendar events through voice or text input, without needing to manually type out details.

Accessibility:

- Individuals with visual or motor impairments can access and manage their calendar more easily through voice commands.

Convenience:

- The chatbot provides a simple and intuitive way to add calendar events, especially for those who are less tech-savvy or prefer a more conversational interface.

Improved organization:

- By easily adding tasks and reminders, users can better manage their schedules, reducing the risk of missing important deadlines or events.

Hands-free interaction:

- Voice input allows users to add calendar events while driving, cooking, or engaged in other activities where their hands are busy.

Business Opportunity

Population of Singapore: 5.917 million people

- *Source: https://www.singstat.gov.sg/find-data/search-by-theme/population/population-and-population-structure/latest-data*

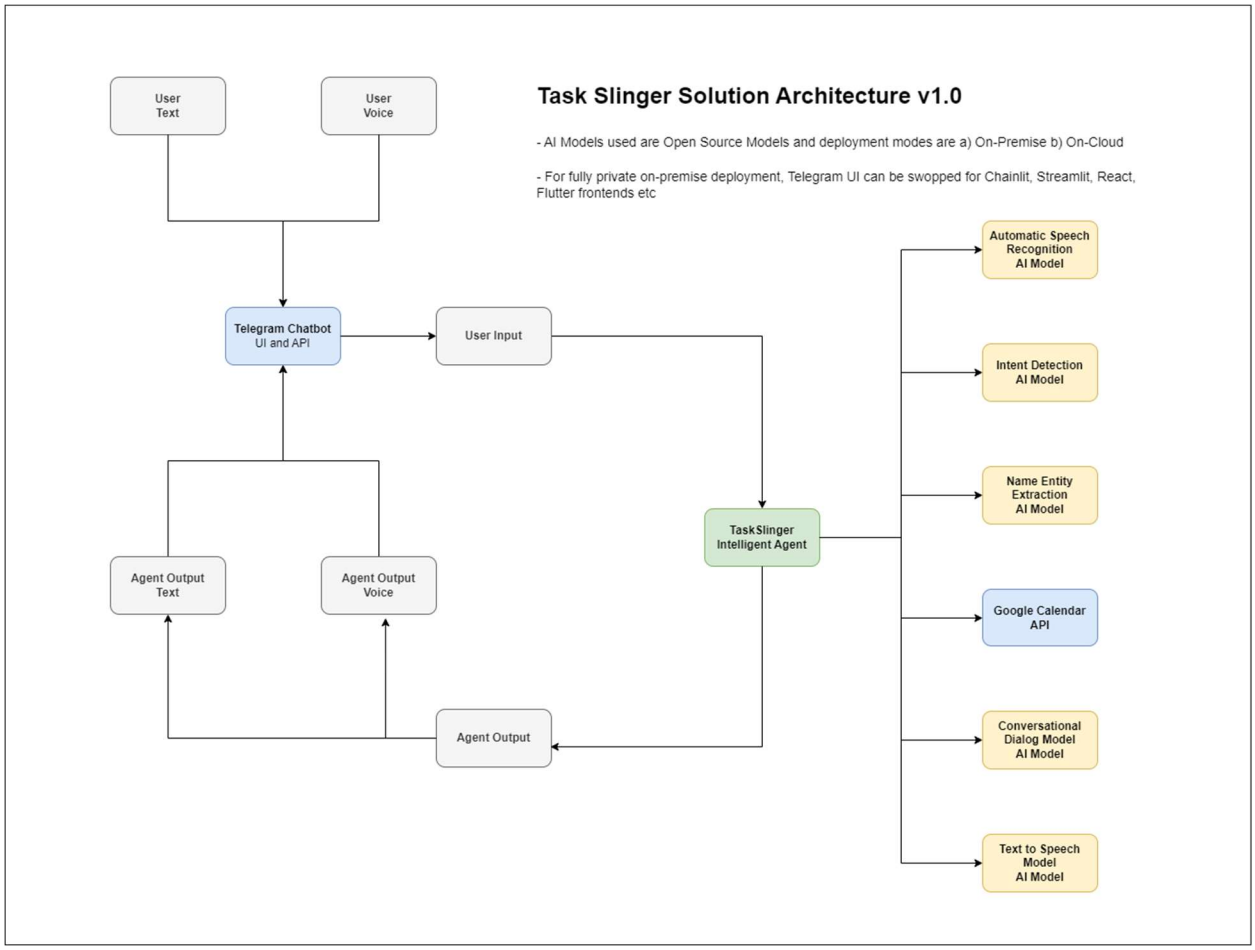
Assuming a subscription price of $1 a month and discounted price of $10 for one year and an adoption rate of 1% of the population, the potential annual revenue is:

- $10 * 0.01 * 5,917,000 = $591,700

Should the pilot be successful, increasing the adoption rate to 2% would make this a million-dollar revenue business. Moreover, TaskSlinger is inherently a global solution that can extended by adding the ability to accept text and voice inputs in new languages. This makes TaskSlinger a potentially viable business.

3.0 SOLUTION

3.1 Scope of POC Solution



The scope of this 10-man-day proof-of-concept to create a Minimum Viable Product that enables the following task flow:

- The user either types or speaks with the TaskSlinger chatbot. If it is speech input, it will be converted to text via the *Automatic Speech Recognition Model*.

- Next, *Intent Detection Model* kicks in to determine what the user wishes to do. If the user wishes to chat, the chatbot will route the user's message to the *Conversational Dialog Model* to generate an appropriate response.

- If the user wishes to add, list, or delete tasks, the *Name Entity Extraction Model* will extract relevant task details such as summary, start date, start time, location, attendees from the user's message.

- Upon successful extraction, TaskSlinger will take the appropriate action via Google Calendar API. In future, this can also be extended to other calendars.
- During interaction with user, if the input is text, TaskSlinger will respond in text. Conversely, if the input is speech, the chatbot will utilise the *Text to Speech Model* to respond with an audio output.

3.2 Functionality Details

3.2.1 Automatic Speech Recognition (ASR)

Model: openai/whisper-small.en

The ASR functionality is obtained from OpenAI's Whisper model. Whisper is a Transformer based encoder-decoder model, also referred to as a sequence-to-sequence model. It was trained on 680k hours of labelled speech data annotated using large-scale weak supervision for automatic speech recognition (ASR) and speech translation.

Model card: https://huggingface.co/openai/whisper-small.en

3.2.2 Intent Detection

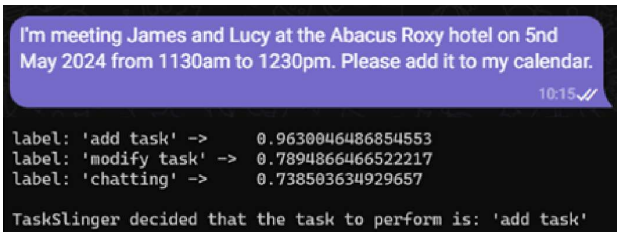
Model: "facebook/bart-large-mnli"

BART is a transformer encoder-decoder (seq2seq) model with a bidirectional (BERT-like) encoder and an autoregressive (GPT-like) decoder. BART is pre-trained by

- Corrupting text with an arbitrary noising function,
- Learning a model to reconstruct the original text.

BART is particularly effective when fine-tuned for text generation (e.g. summarization, translation) but also works well for comprehension tasks (e.g. text classification, question answering).

"Bart-large-mnli" is a Bart based model that has been trained on the MultiNLI (MNLI) dataset.



The Top 3 intents detected are returned. If the scores of the top 2 intents are both above 0.8, a second Language Model is used to confirm the user's intent.

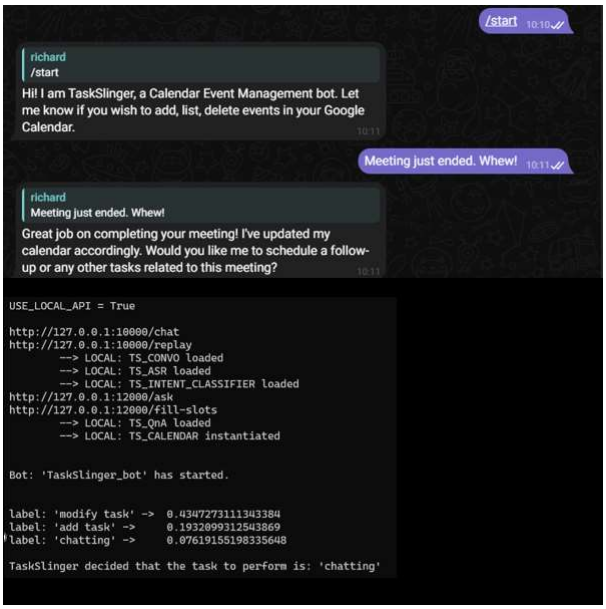
Model card: https://huggingface.co/facebook/bart-large-mnli

3.2.3 Conversational Dialog

Model: "Nous-Hermes-2-Mistral-7B-DPO.Q4"

Nous Hermes 2 on Mistral 7B DPO has good performance improved across the board on all benchmarks tested - AGIEval, BigBench Reasoning, GPT4All, and TruthfulQA. It was trained on primarily synthetic data as well as other high quality datasets, available from the repository teknium/OpenHermes-2.5.

The version used for TaskSlinger is a quantised version GGUF which can run on CPU but with slower performance.



In the above screenshot, TaskSlinger was uncertain what the user's intent was as all the scores were too low. Hence, it did not pick any of the 3 results returned from the Intent Detection Model and used the fall-back intent, which is "chatting".

TaskSlinger does not follow a scripted dialog flow chart. Instead when the user intent "chatting" is detected, the user's message is sent to the Conversational Language Model for completion. The response by the model is then sent to the user.

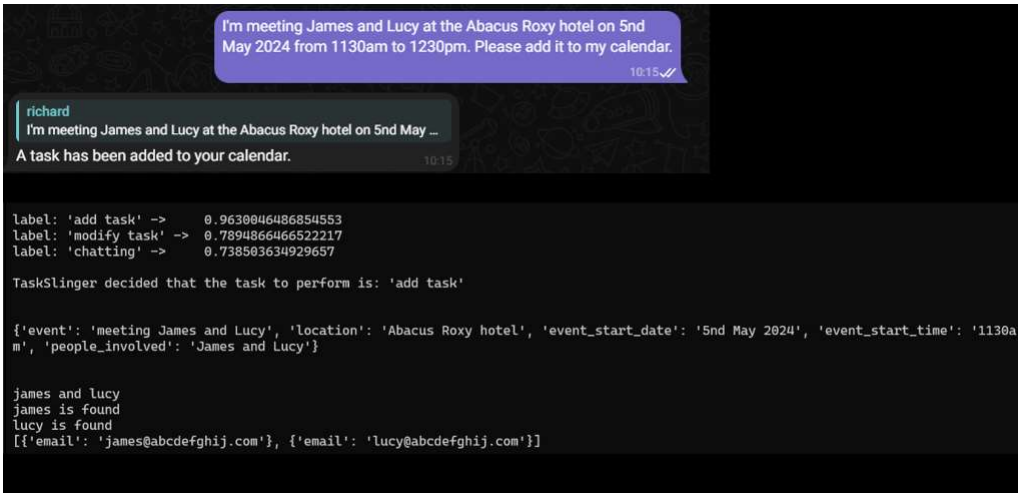
Model card: https://huggingface.co/NousResearch/Nous-Hermes-2-Mistral-7B-DPO

3.2.4 Name Entity Extraction

model='deepset/tinyroberta-squad2'

This is the distilled version of the deepset/roberta-base-squad2 model. This model has a comparable prediction quality and runs at twice the speed of the base model.

As this model has good performance at question and answering, TaskSlinger uses it to identify entities relevant to calendar tasks. While the results still need to be post-processed, the accuracy using this method is quite high and the inference time is fast.



After TaskSlinger determines that the user wants to add a task to the calendar it sends the user message to the Name Entity Extraction Model to extract details that are necessary for task creation in Google Calendar. (screen shot above)

If there are other attendees for the task, TaskSlinger will search for their email address so that they can be inserted in the calendar. In this specific case, the lookup for email addresses was made and it returned two demo email addresses.

Model card: https://huggingface.co/deepset/tinyroberta-squad2

3.2.5 Google Calendar API

Python code is used to communicate with the Google Calendar API to create task, list task, view task details and remove task from the user's calendar.

API details:

- https://developers.google.com/calendar/api/quickstart/python
- https://developers.google.com/calendar/api/guides/overview
- https://github.com/googleapis/google-api-python-client

3.2.6 Text to Speech

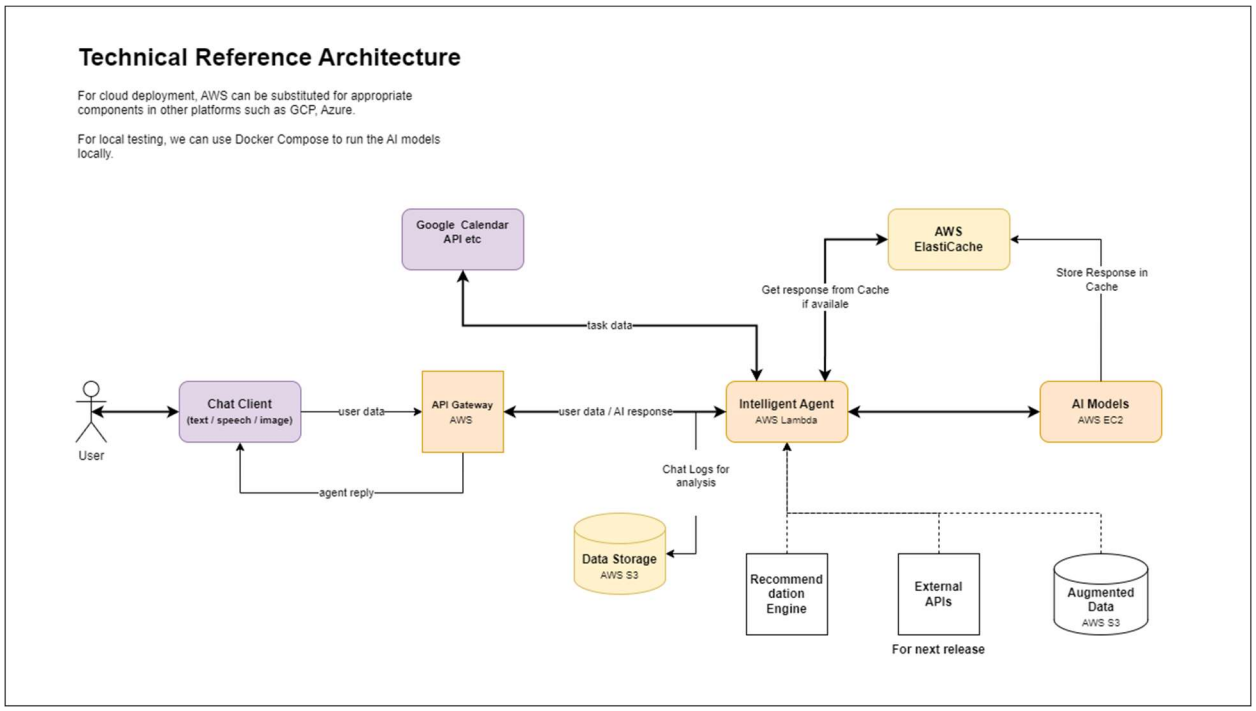
Package: gTTS

gTTS (Google Text-to-Speech), a Python library and CLI tool to interface with Google Translate's text-to-speech API. Writes spoken mp3 data to a file, a file-like object (bytestring) for further audio manipulation, or stdout. It features flexible pre-processing and tokenizing.

Package Details:

- https://gtts.readthedocs.io/en/latest/module.html

4.0 SYSTEM ARCHITECTURE



The AI Models are deployed as docker containers which can be run locally using Docker Compose (if you have sufficiently powerful compute) or run online using AWS, GCP or Azure.

The diagram above shows a technical reference architecture for AWS.

When the user uses a chat client e.g. Telegram to send their message by text or voice, Telegram Server will send that message to our Intelligent Agent TaskSlinger will then process it and respond.

If privacy and confidentiality is important, the front-end chat interface can be swopped for one created by Chainlit, Streamlit, Flutter etc and the AI Models can be deployed on an internal cloud or at the user's desktop/laptop.

The components in white are for a later release where TaskSlinger make recommendations for venues or gifts etc when the user wishes to create a task for a meal or a celebration.

5.0 CONCLUSION / AREA OF IMPROVEMENT

TaskSlinger can bring a lot of benefits to busy people as it removes toil from the mundane work of task management.

It also helps when users are in crowded areas and are unable to easily use their hands to operate digital devices e.g. crowded train or when it is illegal e.g. in a car.

Areas which should be improved in later releases:

- Intent Detection
    - Finetuning of the model with a custom training prompt-pair dataset to be done to improve the ability to detect what the user wants to do.

- Name Entity Recognition and Extraction
    - Finetuning of the model with a custom training prompt-pair dataset to be done to improve the ability to detect entities especially those which are local to the country where TaskSlinger is being used.

- Addition of other Calendaring Software

- To add support for other front end interfaces besides Telegram

- External data

    - Getting latest data from external APIs and augmented data source to make recommendations to the user on venues, venue operating details and gifts for purchase and the creation of a recommendation engine.

- AI Safety and Ethics
    - TaskSlinger should not create tasks in the user's calendar that are illegal, unsafe, or unethical.
    - All AI Models used in this project should be human aligned for safety and ethics.

6.0 BIBLIOGRAPHY & APPENDIX

APPENDIX 1 – Project Proposal

- https://github.com/atsui888/Intelligent-Software-Agents/tree/main/Project_Reports/03%20Other%20Docs

APPENDIX 2 – Installation and User Guide

- https://github.com/atsui888/Intelligent-Software-Agents/tree/main/Project_Reports/03%20Other%20Docs

APPENDIX 3 – System Code

- https://github.com/atsui888/Intelligent-Software-Agents/tree/main/Code