# Proposal

## Background

The project was inspired from a common problem working class adults in Singapore's Central Business District (CBD) faced - where and what to eat for lunch. Daily, the working crowd in CBD look forward to their meal times to recharge and perhaps to catch up with fellow colleagues. However, given the numerous food choices and a lack of time to research on where and what to eat, most people have difficulty deciding their meals. Some have resorted to generating random places to eat using Microsoft Excel. This project aims to suggest places to eat through a chatbot, in order to provide an interactive user experience and to enable the user to make a more informed choice. The chatbot will be built using Natural Language Processing (NLP) techniques.

## Motivation

Chatbots, through Natural Language Processing (NLP), is an attractive method of human-computer interaction due to the interactivity as compared to traditional mediums such as a Frequently Asked Questions (FAQ) section on a website. It has been used in a variety of domains such as environment and sustainable development [1], education [2] and other commercial purposes [3]. The fact that chatbots are not widely used in Singapore for food and restaurants recommendations makes it an obvious primary choice for us to develop one which is specific to Singapore. Furthermore, creating relevant and appropriate response leveraging on various NLP techniques is a tough but critical component to a successful chatbot. Hence, we decided to take on the challenge of developing one. There are also several use cases or scenarios which we think will be immediately useful. They are:

* Tying up with business to provide exposure as well as a source of revenue (e.g. incentives/discounts)
* Letting users to subscribe to this service to receive information about trending restaurants based on their preferences
* Providing some entertainment (localized humour) when choosing where to eat
* Providing other information about food location by complementing current data with other data sources such as Google visitor information (e.g. Popular time, live visits, duration)
* Generating revenue through advertisements within App

## Analytics Tasks

Here we describe the analytics tasks involved in building the chatbot

#### Named Entity Extraction

* Extracting foods, places and locations from user's comments

#### Document Retrieval

* Use of document retrieval to get the most relevant place to have lunch

#### Topic Analysis

* Reviews are used to train the bot to provide relevant responses.
* Each review is tied to a topic

## Approach

* Get the data (APIs and scraping)
* Prepare the corpus
* Prepare statement / response pairs?
* Use part of speech tags to understand?
* Use state machines to model the states
* Use information retrieval to get responses
* Use comparison model: cosine similarity to return relevant dictionary
* Use generic grammar rules to construct responses

## Data

#### Data Preparation

* web scraping
* nlp processing (vector space)
* training - topic modeling etc

#### Initial Exploration of Scraped Data

* Some description and charts
  + length of reviews
  + number of businesses
  + number of foods
  + number of places
  + number of cuisines

## Proposed Bot Architecture

### Python Classes

* jiakbot - main bot class
* parsing - tokenization
* state machine - entity extraction
* responder - construct responses
  + The main Responder class handles all response construction.
  + Points the flow in two main directions:
    - Information retrieval from the database
    - Construct an alternative response if a non-food query is detected
  + The Responder class makes use of several other classes to handle greetings, and other non-food related topics that are detected in the conversation
* retriever - information extraction
* The Retriever function is meant to query the database based on the data provided by the Responder and return a relevant set of information for the Responder class to utilize
* The Retriever function conducts 4 main steps:
  + Simple SQL statements used to extract the related data from the database based on data (query) provided by Responder
  + Reference the tokens: Compare using cosine similarity to get most relevant statement
  + Convert set of statement(s) into dictionary
  + Return dictionary to Responder class
* trainer/topic\_finder - topic modeling

### Database Diagram

* insert database diagram

### References

[1] AluxBot - A Chatbot that Encourages the Care for the Environment. (2016). International Journal of Computer Science Issues, 13(6), 120-123. [2] Yi Fei Wang, & Stephen Petrina. (2013). Using Learning Analytics to Understand the Design of an Intelligent Language Tutor – Chatbot Lucy. International Journal of Advanced Computer Science and Applications, 4(11), 124-131. [3] Chatbots Raise Questions About the Future of Customer Service. (2016, April 27). PR Newswire, p. PR Newswire, Apr 27, 2016.