C3879 – Capstone Project
Yelper Assistant

PROJECT PRESENTATION

03-AUG-2019

Presented By:
Lim Yuan Her (17060167)

Agenda

1. **Business Analysis**
   - ❑ Problem/ Solution
   - ❑ System Functional Requirements

2. **Schedule**
   - ❑ Tasks/ Milestones

3. **System Design**
   - ❑ Overall Architecture
   - ❑ Use Cases
   - ❑ Implementation/ Deployment

4. **Recommender Design**
   - ❑ Dataset
   - ❑ Model Evaluation
   - ❑ Recommender Workflow

5. **Chatbot Design**
   - ❑ Interface
   - ❑ Converrsation Design
   - ❑ Intents/Entities/Contexts
   - ❑ Fulfilment/ Integration

6. **Future Enhancements**

Business Case

❑ Problem

- Yelp is a search service using crowd-sourced reviews about local businesses. Besides reviews, it also facilitates searching for events, lists and communication between Yelp users

- *Ease of access* – uses traditional menu type access whereby the user needs to search for the exact option to fulfill his/her search requirement, which is tedious and time-consuming..

- *Results Differentiation* – provides generic results that may or may not be relevant to the user. This will turn off the user who will search for other sites that can provide customized results.

- *User Engagement* – site content layout adopts a traditional approach more suited for a desktop browser experience and lacks the personalized touch associated with modern interfaces.

❑ Solution Overview

- 2 major features to address identified problems:

  - Incorporate a virtual assistant to provide a natural language interface for interaction with the system, thus reducing time spent searching for relevant menu/ option.

  - Incorporate a recommender engine to provide personalized recommendations based on a user's previous rating behavior and similarity to other users

Business Analysis

- **System Functional Requirements / Implementation**
  - Provide a recommendation engine to provide personalized recommendation
  - Provide a virtual assistant with social media integration e.g. Slack

Project Schedule

❑ Project Implementation

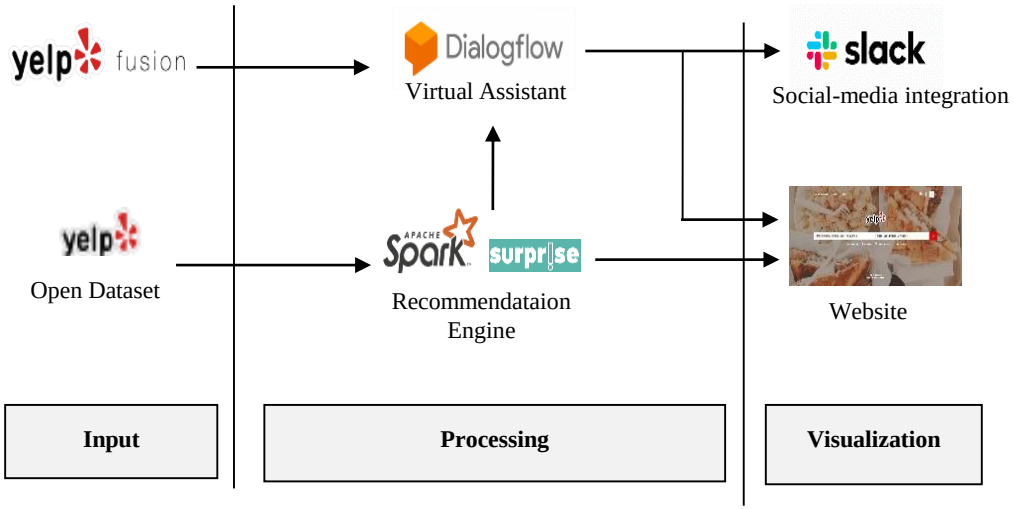
- 7 main phases
  - Data Cleaning/ Preparation
  - Recommender Algorithm Analysis and Evaluation
  - Recommender Development
  - Chatbot Development
  - Recommender/Chatbot Integration
  - Functional Testing
  - Documentation (Report + Presentation)

Task	Week No														
	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Data Cleaning/ Exploration															
Recommender algorithm analysis and evaluation															
Recommender Development															
Chatbot Development															
Recommender/ Chatbot Integration															
Testing															
Demo Preparation/ Presentation															

System Design – Overall Architecture

❑ Overall System Architecture

- **Input**
  - Yelp Open Dataset / Yelp Fusion API
- **Processing**
  - DialogFlow (Chatbot)
  - Scikit-Surprise (Recommender)
- **Visualization**
  - Slack/ Telegram, Website



System Design – Use Cases

- **Search Business**
  - Category – category filters e.g. Bike Rental, Bakeries, breweries etc.
  - Name
  - Phone
  - Price - Price Level indicator e.g. 1 for $, 4 for $$$$, 2,3 for $$ and $$$ etc.
  - Business ID

- **Search Businesses with Food Delivery services**
  - search for a business served by food delivery services

- **Search for Businesses with specific characteristics**
  - *Hot and New* - popular businesses which recently joined Yelp
  - *Cashback* - businesses offering Yelp Cash Back to in-house customers
  - *Deals* - businesses offering Yelp Deals on their profile page
  - *Wheelchair Accessible* - businesses which are wheelchair accessible
  - *Reservation* - businesses with Yelp Reservations bookings enabled on their profile page
  - *Waitlist Reservation* - businesses with Yelp Waitlist bookings

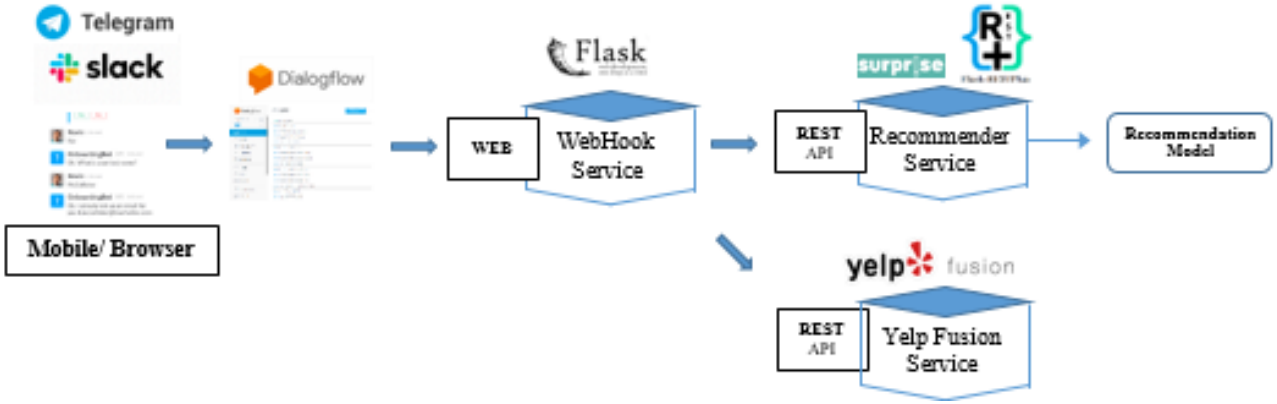
- **Get Recommendations**
  - search for customized recommendation based on Yelp user ID

- **Featured Events**
  - search for events as chosen by Yelp's community managers

System Design - Implementation/ Deployment

❑ Microservices architecture

- *WebHook Service* – Fulfilment engine for DialogFlow webhook hosted on Heroku platform at https://yelperassistant-wh.herokuapp.com
- *Recommender Service* – Recommendation engine using Scikit-Surprise for serving customized recommendations hosted on Heroku platform at https://yelperassistant-rec.herokuapp.com/api/v1
- *Yelp Fusion API Service* – Directory service using Yelp Fusion API for accessing Yelp directory information at https://api.yelp.com



❑ Interface

- DialogFlow chatbot integrated with:
  - Slack workspace at https://yelperassistant.slack.com via web browser or slack mobile app
  - Telegram chatbot

Recommender Design - 1

❑ Dataset

- Yelp Open Dataset (https://www.yelp.com/dataset)

Name	Description	Size
business	Contains business information e.g. address, attributes, operating hours etc.	192K
review	Contains review ratings for businesses by users	6.7M
user	Contains user information e.g. name, # of reviews, average rating etc.	1.6M

- To facilitate development, 20K review data were extracted to minimize the computational costs involved

❑ Model Evaluation

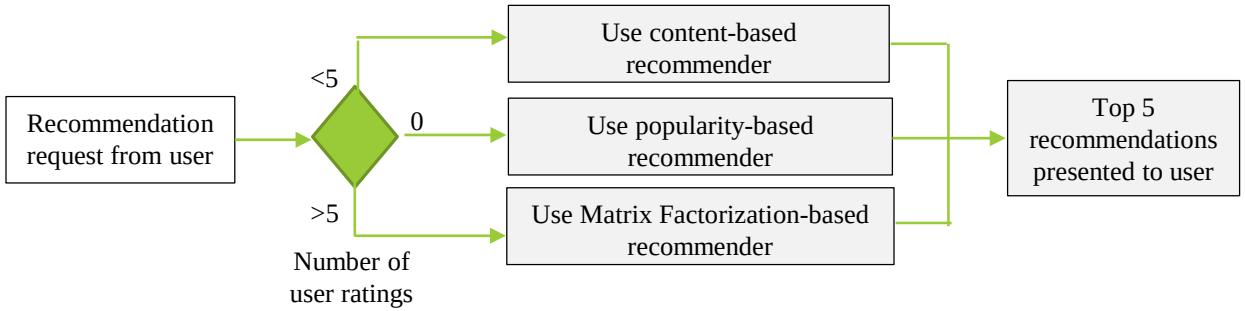
- SVD++ algorithm outperformed all others

Algorithm	Description	RMSE
KNNBasic	Basic collaborative filtering algorithm	1.4678
KNNWithMeans	Basic collaborative filtering algorithm, taking into account the mean ratings of each user	1.4906
KNNWithZScore	Basic collaborative filtering algorithm, taking into account the z-score normalization of each user	1.4906
SVD	SVD algorithm popularized by Simon Funk during the Netflix Prize	1.4344
SVDpp	SVD++ algorithm (extension of SVD) taking implicit ratings into account	1.4309
NMF	Collaborative filtering algorithm based on Non-negative Matrix Factorization	1.5043
ALS	Matrix factorization algorithm that uses Alternating Least Squares with Weighted-Lamda-Regularization (ALS-WR)	3.9320

Recommender Design - 2

❑ Recommender Workflow

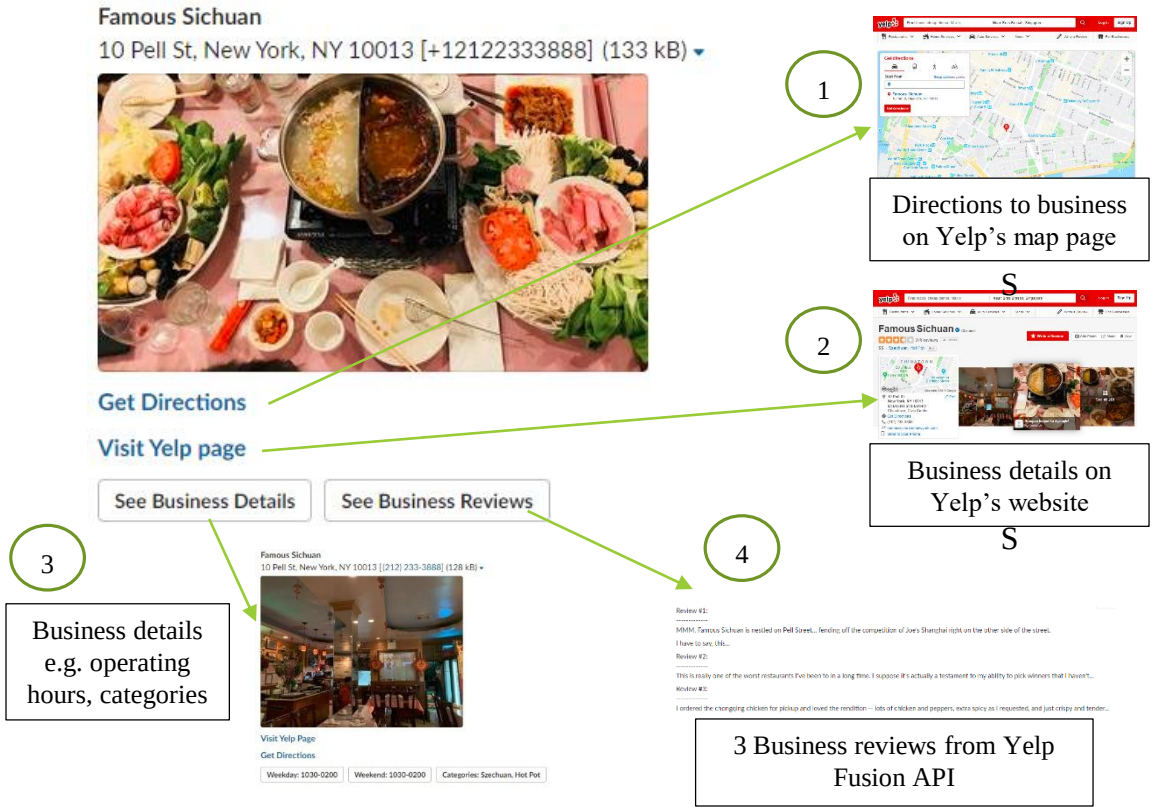
- ➢ To facilitate development, 20K review data were extracted to minimize the computational costs involved



Recommender Type	Description	Criteria
Popularity-based	▪ Provides top 5 businesses with highest number of user ratings and average rating value ▪ Assumes no user information is available	For new users
Content-based	▪ uses business "categories" attribute to calculate similarity scores between businesses using cosine similarity measure ▪ Top 5 similar businesses for each business rated by user is combined and sorted by aggregated count and the top 5 business (by count) is recommended	For users with <5 ratings
Matrix Factorization-based	▪ Uses SVD++ algorithm to capture the relationship between users and businesses using a latent factor model ▪ Uses SVDpp class from Scikit-Surprise package	For users with >5 ratings

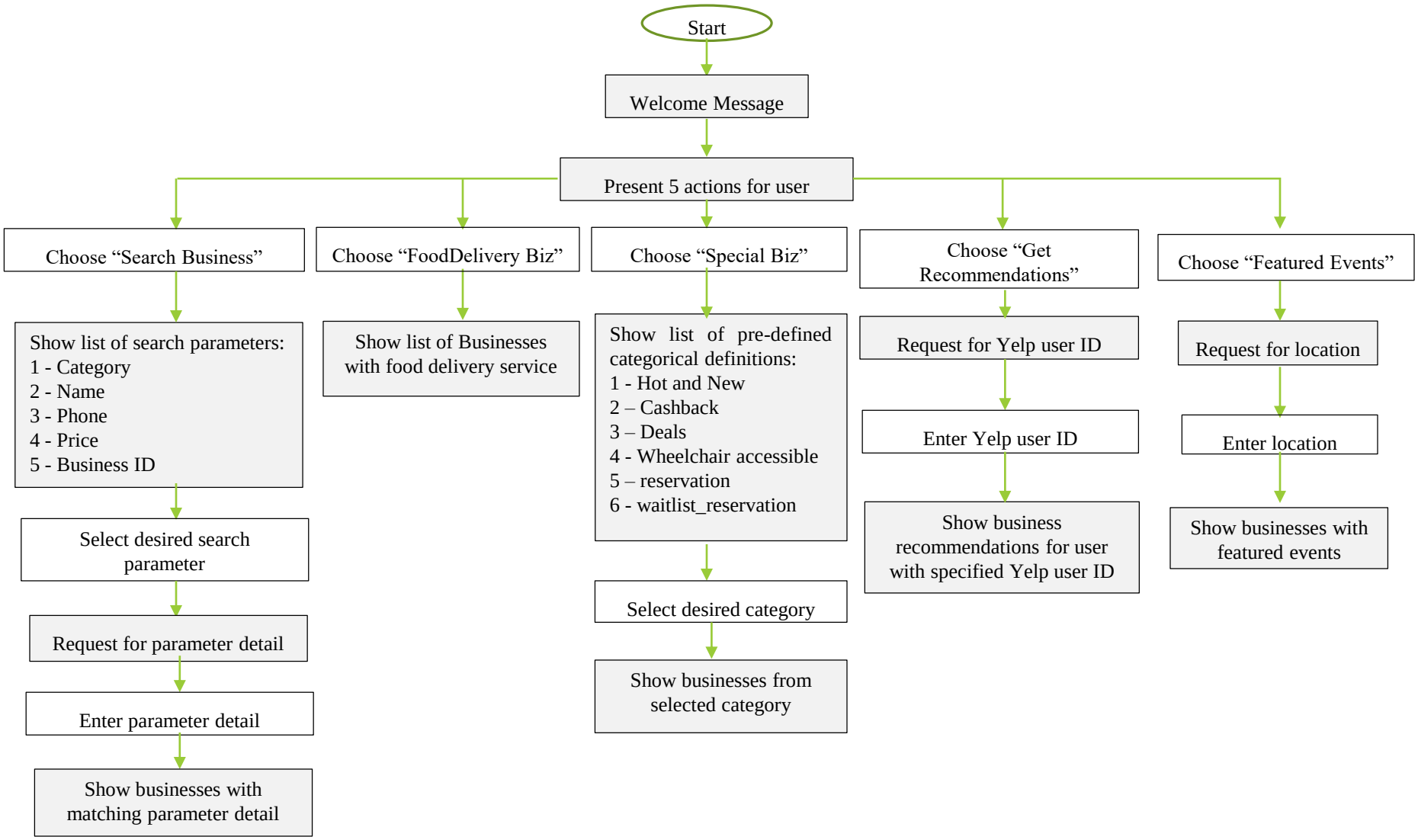
Chatbot Design - 1

- **Interface**
  - Uses Google DialogFlow as the natural language processing engine
  - Q&A type to provide quick digestible search results with navigable links to more detailed information if required



Chatbot Design - 2

- **Conversation Design**
  - main conversation flow using buttons where appropriate to minimize input error



Chatbot Design - 3

❑ Intents

Action	Intent	Purpose	Parameters
-	Default.Welcome	Greet user and offer services	-
-	Default.Fallback	Handles unexpected users' responses	-
-	General.ConnectionTest	Tests connection to webhook service	-
getBusinessReviews	Directory.getBusinessReviews	Gets 3 reviews for specified business	businessID
getFeaturedEvents	Directory.getFeaturedEvents	Gets 3 featured events for specified location	latitude, longitude, location
matchBusiness	Directory.matchBusiness	Finds matching businesses with search parameters	See (1)
	Directory.searchBusiness	"Search Business" parameter listing	
searchBusiness	Directory.searchBusinessByAttribute	Gets businesses with matching attribute	See (1)
searchBusiness	Directory.searchBusinessByCategory	Gets businesses in specified category	See (1)
getBusiness	Directory.searchBusinessByID	Gets businesses with specified ID	businessID
searchBusiness	Directory.searchBusinessByName	Get business with matching name	See (1)
searchBusinessByPhone	Directory.searchBusinessByPhone	Gets business with matching phone #	phone
searchBusiness	Directory.searchBusinessByPrice	Gets business with matching price ranges	See (1)
searchFoodDeliveryBusinesses	Directory.searchFoodDeliveryBusinesses	Gets buseinsses with food delivery services	latitude, longitude, location
getRecommendations	Rec.getRecommendations	Gets recommendations for specified user	user_id

Chatbot Design - 4

Entities

Entity	Description	Examples
categories	List of categories for "Directory.searchBusinessByCategory" intent	bikerentals, dentists, chiropractors, beaches, bbq

Context

- Input context "searchBusiness" assigned to control the conversation flow to start from Directory.searchBusiness intent
  - Directory.searchBusinessByCategory
  - Directory.searchBusinessByName
  - Directory.searchBusinessByPhone
  - Directory.searchBusinessByPrice

Fulfilment

- Webhook developed as Flask web application
- Hosted on https://yelperassistant-wh.herokuapp.com

Integration

- Integrated with Slack and Telegram
- Slack messaging workspace at https://yelperassistant.slack.com

- Telegram at https://web.telegram.org/#/im?p=@YelperAssistantbot

Further Enhancements

- Integrating with other social messaging platforms e.g. Facebook Messenger, LINE etc. and physical virtual assistant devices e.g. Google Assistant, Amazon Alexa etc.
- Perform sentiment analysis based on business textual reviews for ratings verification
- Use Baynesian ranking for popularity-based recommender model to take into account number of ratings and average rating given for specific business instead of current simple aggregated count sorting method
- Perform anomaly detection for detecting shilling attacks (manipulation of recommendation rankings)
- Integrate Yelp's security framework to facilitate business rating/review directly from chatbot

Q&A