Development of a Digital Assistant for Travel Booking - Phase I

By: Anthony Leung

Agenda

1. Overview
2. Methodology & Results
    a. Proof of Concept Model
3. Development Demo
4. Summary and Next Steps

1

OVERVIEW

Product Introduction
Development Plan

Product Introduction

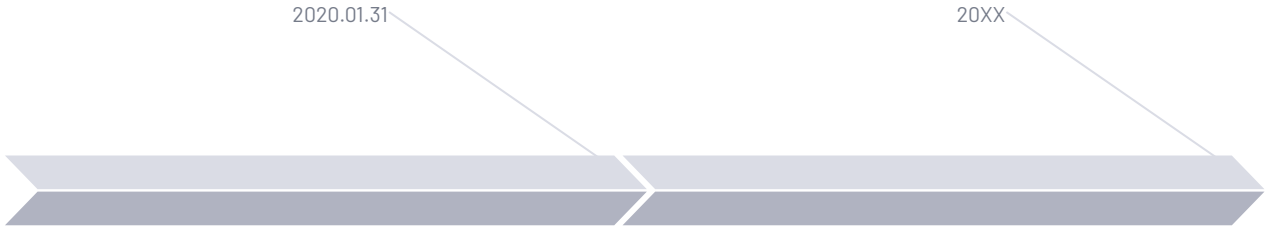
NotSnapTravel wants to create a digital assistant (i.e., chatbot) that allows users to query the following through conversation:

FLIGHT PRICES

HOTEL BOOKINGS

CAR RENTALS

Development Plan



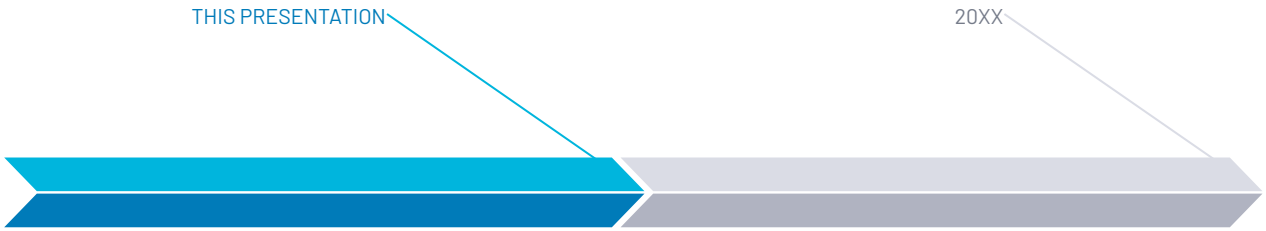
Phase 1

Create a proof of concept model that can extract relevant flight information from conversational text.

Phase 2

Interface model with an API and deploy it in a suitable platform (e.g., Slack).

Development Plan



Phase 1

Create a proof of concept model that can extract relevant flight information from conversational text.

Phase 2

Interface model with an API and deploy it in a suitable platform (e.g., Slack).

2

METHODOLOGY & RESULTS

Natural Language Understanding
NLP Transfer learning (ULMFit)
Named Entity Recognition & Phrase Matching

Development Toolbox

Experiment/
Prototype
environment

Data
manipulation

Deep learning
modeling

NLP

?

How does the chatbot know what to respond to the user?

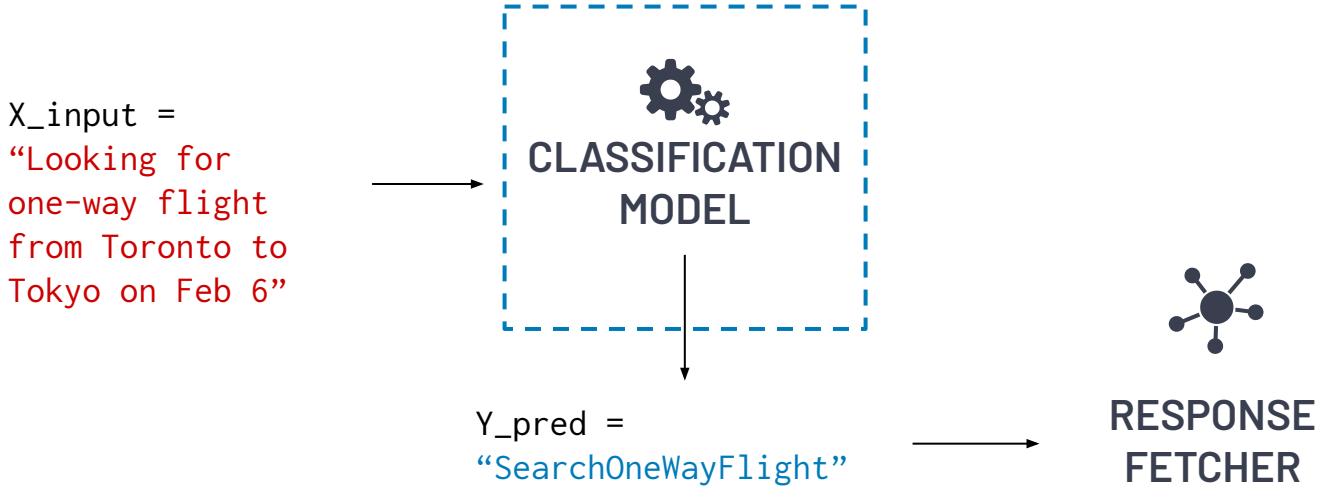
NLU: What is the intent when we say stuff?

- Natural Language Understanding:
  The ability to interpret text.

SPEECH: “I am looking for one-way flight from Toronto to Tokyo, leaving on Feb 6”

INTENT: SearchFlightPrice

Classification: Associating speech with intents



Dataset size of 236 across 8 different intents

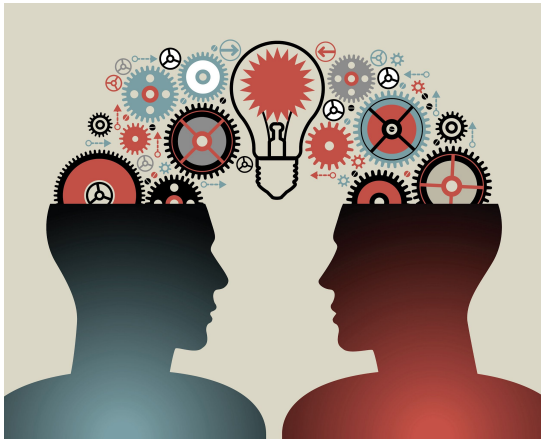
Intent	Count
SearchOneWayFlight	41
SearchHotelPrice	37
SearchCarRentalPrice	35
Greeting	33
Goodbye	26
GiveThanks	25
SearchRoundTripFlight	24
Help	15
Total	**236**

Challenge: Obtaining an accurate model with a small corpus of travel queries.

How can we obtain one in a feasible manner?

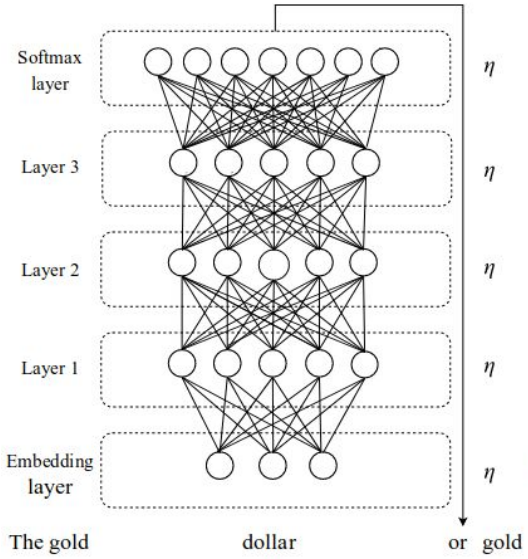
NLP Transfer Learning: ULMFit

- Universal Language Model Fine-tuning (ULMFit)**[*]**
- A transfer learning approach to text classification.
- Beneficial for tasks with small datasets (i.e., it's fast with great accuracy!).
- Can be perform on fastai.



[*] J. Howard & S. Ruder, *arXiv preprint:1801.06146* (2018).

NLP Transfer Learning: ULMFit



(a) LM pre-training

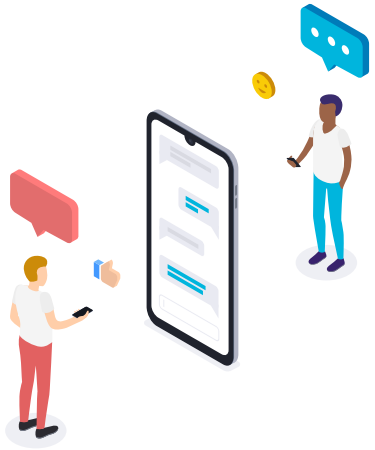
[*] J. Howard & S. Ruder, *arXiv preprint:1801.06146* (2018).

Training Datasets for LM and NN

a) LM Pre-training:
- Utilize the **AWD-LSTM[**]** pretrained model.

b) LM Fine-tuning:
- "General-Chat-Log-Telegram" (13,207 sentences) + Present dialogues w/o intents (236 sentences).

c) Classification fine-tuning:
- Present dialogues w/ intents (236 sentences).

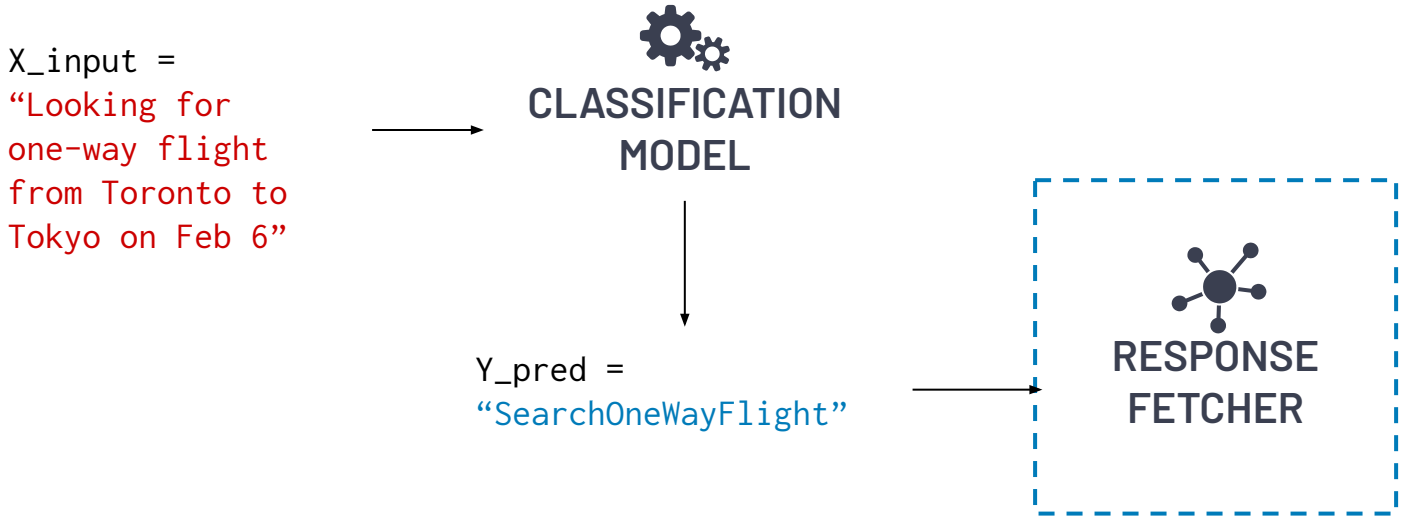


[**] S. Merity, N. S. Keskar, R. Socher, *arXiv preprint:1708.02182* (2017)

Overall intent classification accuracy: 91%

Intent	Accuracy
SearchRoundTripFlight	100%
SearchCarRentalPrice	100%
Greeting	100%
Goodbye	100%
SearchOneWayFlight	92%
SearchHotelPrice	83%
Help	83%
GiveThanks	78%

Chatbot blueprint



?

How can the chatbot recognize locations and dates from text?

Named entity recognition (NER): Recognizing places, dates, etc.

- Utilize spaCy library.
- E.g., “Looking for one-way flight from Toronto to Tokyo on Feb 12”

One CARDINAL -way flight from Toronto GPE to Tokyo GPE on Feb 12 DATE

	text	start	end	label
0	One	0	3	CARDINAL
1	Toronto	20	27	GPE
2	Tokyo	31	36	GPE
3	Feb 12	40	46	DATE

Which one is origin and destination?

?

How can the chatbot distinguish between origin and destination?

Phrase Matcher: Match sequence of tokens based on pattern rules

- Can combine Parts-of-Speech tags and NER labels.

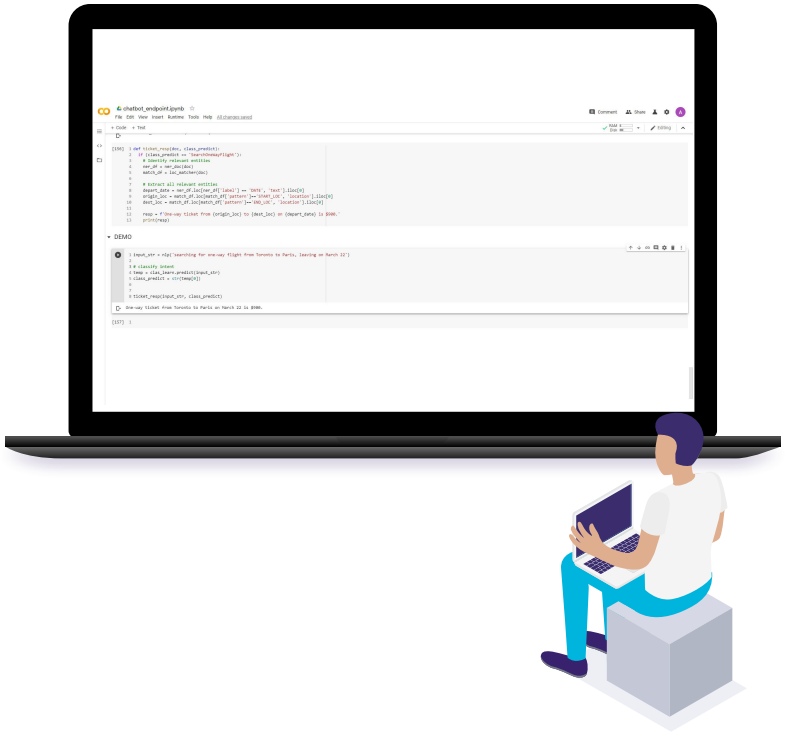
```
# Pattern: “from [this loc]”
start_loc = [{‘LOWER’: ‘from’},
             {‘ENT_TYPE’: ‘GPE’}
            ]
# Pattern: “to [that loc]”
end_loc = [{‘LOWER’: ‘to’},
           {‘ENT_TYPE’: ‘GPE’}
          ]
```

	pattern	start	end	text	location
0	START_LOC	4	6	from Toronto	Toronto
1	END_LOC	6	8	to Tokyo	Tokyo

3

DEVELOPMENT DEMO

(Subject to Changes)



4

SUMMARY AND NEXT STEPS

Summary and Next Steps

- Transfer learning (ULMFiT) was used to train a neural network that can classify speech intent.
- Relevant entities of travel queries are extracted by using NER and can be differentiated by Phrase Matching.
- **Next Steps:**
  - Further improve the speech intent classifier by including more dialogue and fine-tuning hyperparameters.
  - Interact chatbot model with a travel API.
  - Deploy chatbot onto a platform (e.g., Slack).