Task-oriented dialog systems

Task-oriented dialog system

You can talk to a personal assistant:

- Apple Siri
- Google Assistant
- Microsoft Cortana
- Amazon Alexa
- …

You can solve these tasks:

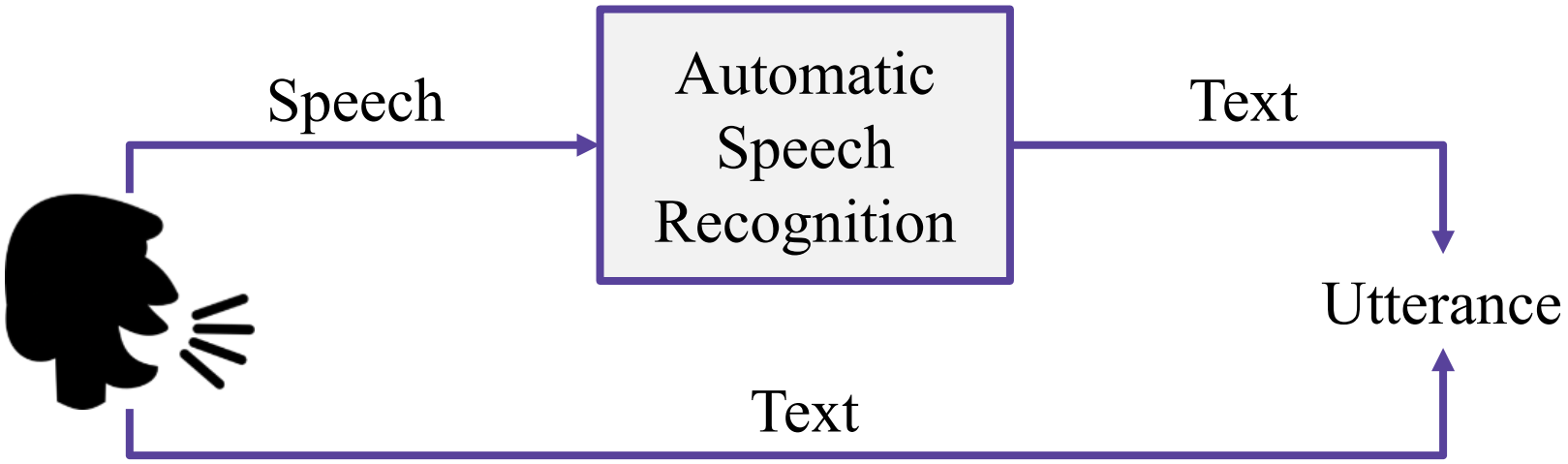
- Set up a reminder
- Find photos of your pet
- Find a good restaurant
- Send a message
- …

Task-oriented dialog system

You can write to a chat bot:

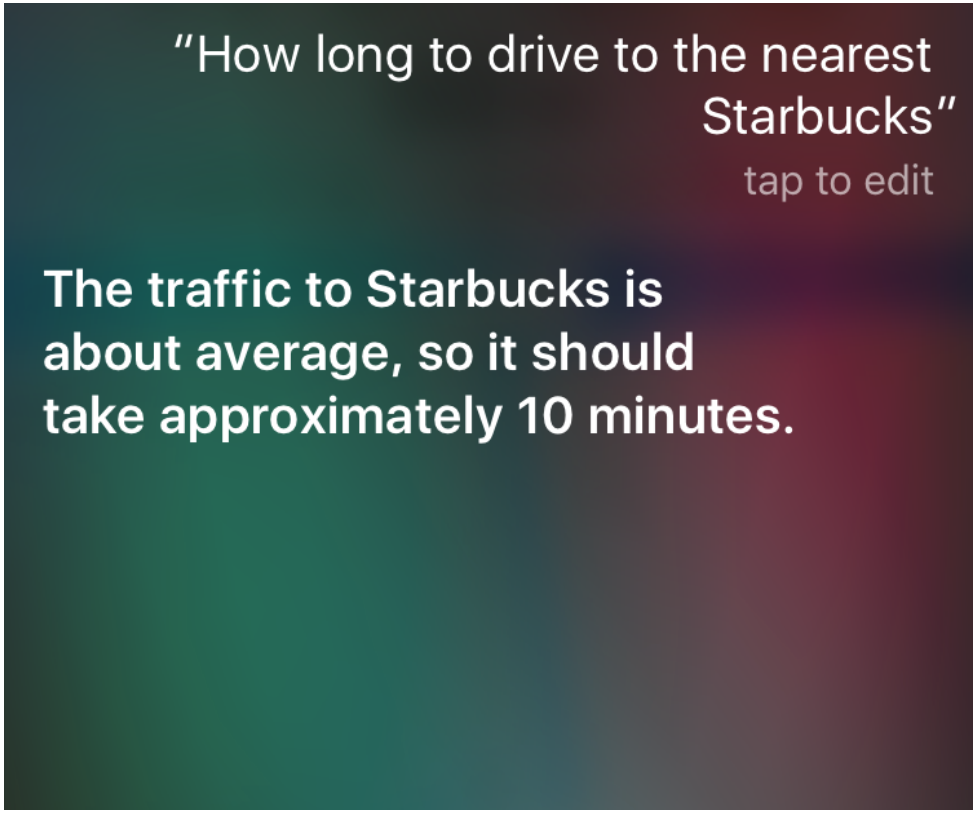
- To book tickets
- To order food
- To contest parking tickets
- To track expenses
- …

Utterance



Intent classification

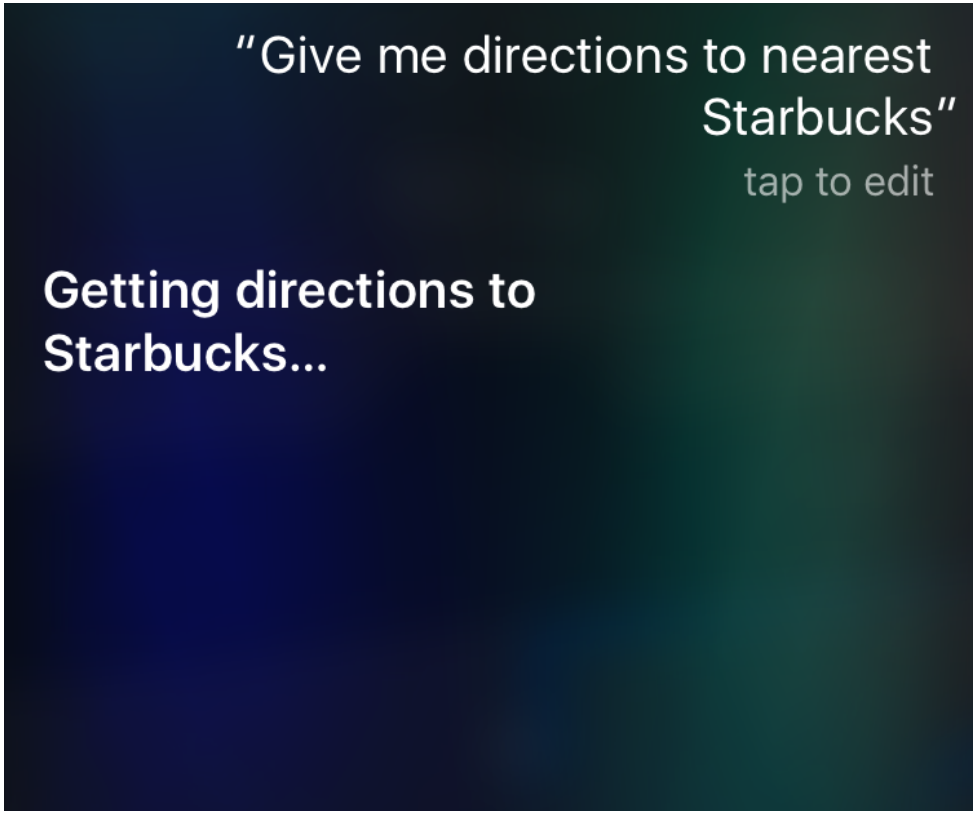
- What does the user want?
- Which predefined scenario is the user trying to execute?



Intent: **nav.time.closest**

There’re many intents

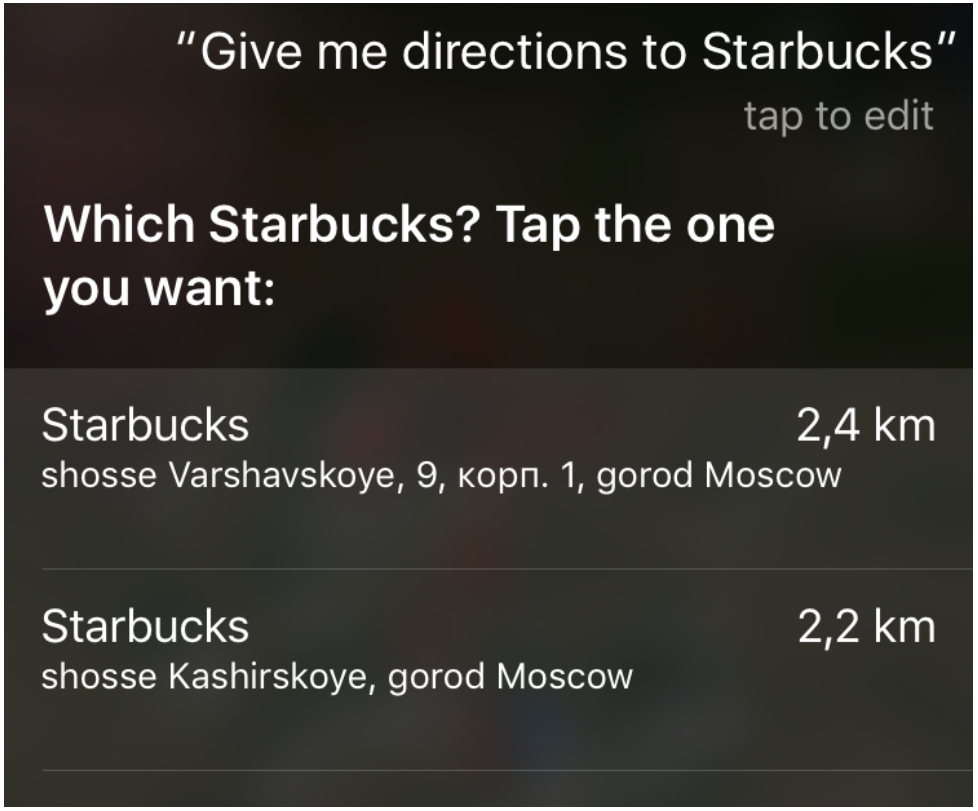
- And you need to classify them to give correct answers
- This is a classification task and you can measure **accuracy**



Intent: **nav.directions.closest**

And one more example

- This time assistant needs additional information and initiates dialog



Intent: **nav.directions**

Form filling approach to dialog management

- Think of an intent as a **form** that a user needs to fill in.
- Each intent has a set of fields (**slots**) that must be filled in to execute the request.

- Example: **nav.directions** intent
  - **@FROM** slot: defaults to current geolocation
  - **@TO** slot: required

- We need a **slot tagger** to extract slots from utterance.

Slot filling/tagging

- Example:
  - User: Show me the way to History Museum
  - Slot tagger: Show me the way to @TO{History Museum}
- This is token classification task in **BIO** scheme:
  - **B** corresponds to the word at the **beginning** of the slot
  - **I** corresponds to the word **inside** the slot (excluding beginning)
  - **O** corresponds to words **outside** of slots
- Example of **BIO** scheme tagging:

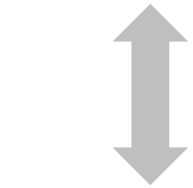
Tokens	Show	me	the	way	to	History	Museum
Tags	O	O	O	O	O	B-TO	I-TO

Slot filling/tagging

- You **train** it as a sequence tagging task in BIO scheme
- A slot is considered to be correct if its range and type are correct
- $\textbf{Recall} = \frac{\textbf{\# correct slots found}}{\textbf{\# true slots}}$
- $\textbf{Precision} = \frac{\textbf{\# correct slots found}}{\textbf{\# found slots}}$
- You can **evaluate** slot tagger with $\mathbf{F_1} = \mathbf{2} \times \frac{\textbf{Precision} \times \textbf{Recall}}{\textbf{Precision} + \textbf{Recall}}$

Form filling dialog manager (single turn)

- User: **Give me directions to San Francisco**
  - Intent classifier: nav.directions
  - Slot tagger: @TO{San Francisco}
  - Dialog manager: *all slots are filled, here's the route*
- Agent (assistant): **Here's the route**



Google Maps

Form filling dialog manager (multi-turn)

- User: **Give me directions from Los Angeles**
  - Intent classifier: nav.directions
  - Slot tagger: @FROM{Los Angeles}
  - Dialog manager: *required slot is missing, where to?*
- Agent (assistant): **Where do you want to go?**
- User: **San Francisco**
  - Intent classifier: nav.directions
  - Slot tagger: @TO{San Francisco}
  - Dialog manager: *okay, here's the route*
- Agent (assistant): **Here's the route**

Google Maps

Form filling dialog manager (multi-turn)

- User: **Give me directions from Los Angeles**
  - Intent classifier: nav.directions
  - Slot tagger: @FROM{Los Angeles}
  - Dialog manager: *required slot is missing, where to?*
- Agent (assistant): **Where do you want to go?**
- User: **San Francisco**
  - Intent classifier: nav.directions
  - Slot tagger: @TO{San Francisco}
  - Dialog manager: *okay, here's the route*
- Agent (assistant): **Here's the route**

Google Maps

Form filling dialog manager (multi-turn)

- User: **Give me directions from Los Angeles**
  - Intent classifier: nav.directions
  - Slot tagger: @FROM{Los Angeles}
  - Dialog manager: *required slot is missing, where to?*
- Agent (assistant): **Where do you want to go?**
- User: **San Francisco**
  - Intent classifier: nav.directions
  - Slot tagger: @TO{San Francisco}
  - Dialog manager: *okay, here's the route*
- Agent (assistant): **Here's the route**

We need context here

Google Maps

How to track context (an easy way)

- Both intent classifier and slot tagger need context (what happened before)

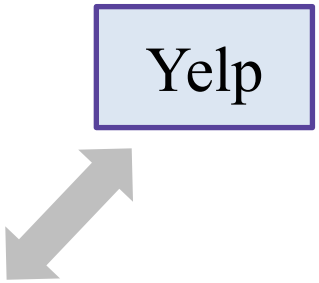
- Let's add simple **features** to both of them:
  - Previous utterance intent as a categorical feature
  - Slots filled in so far with binary feature for each possible slot
- Improves slot tagger F1 by 0.5%
- Reduces intent classifier error by 6.7%

- A better way: memory networks

http://www.cs.toronto.edu/~aditya/publications/contextual.pdf

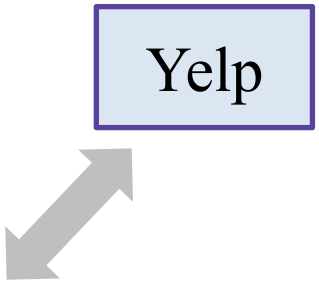
How to track a form switch

- User: **Give me directions from Los Angeles**
  - Intent classifier: nav.directions
  - Slot tagger: @FROM{Los Angeles}
  - Dialog manager: *required slot is missing, where to?*
- Agent (assistant): **Where do you want to go?**
- User: **Forget about it, let's eat some sushi first**
  - Intent classifier: nav.find
  - Slot tagger: @CATEGORY{sushi}
  - Dialog manager: *okay, let's start a new form and find some sushi*
- Agent (assistant): **Okay, here are nearby sushi places**

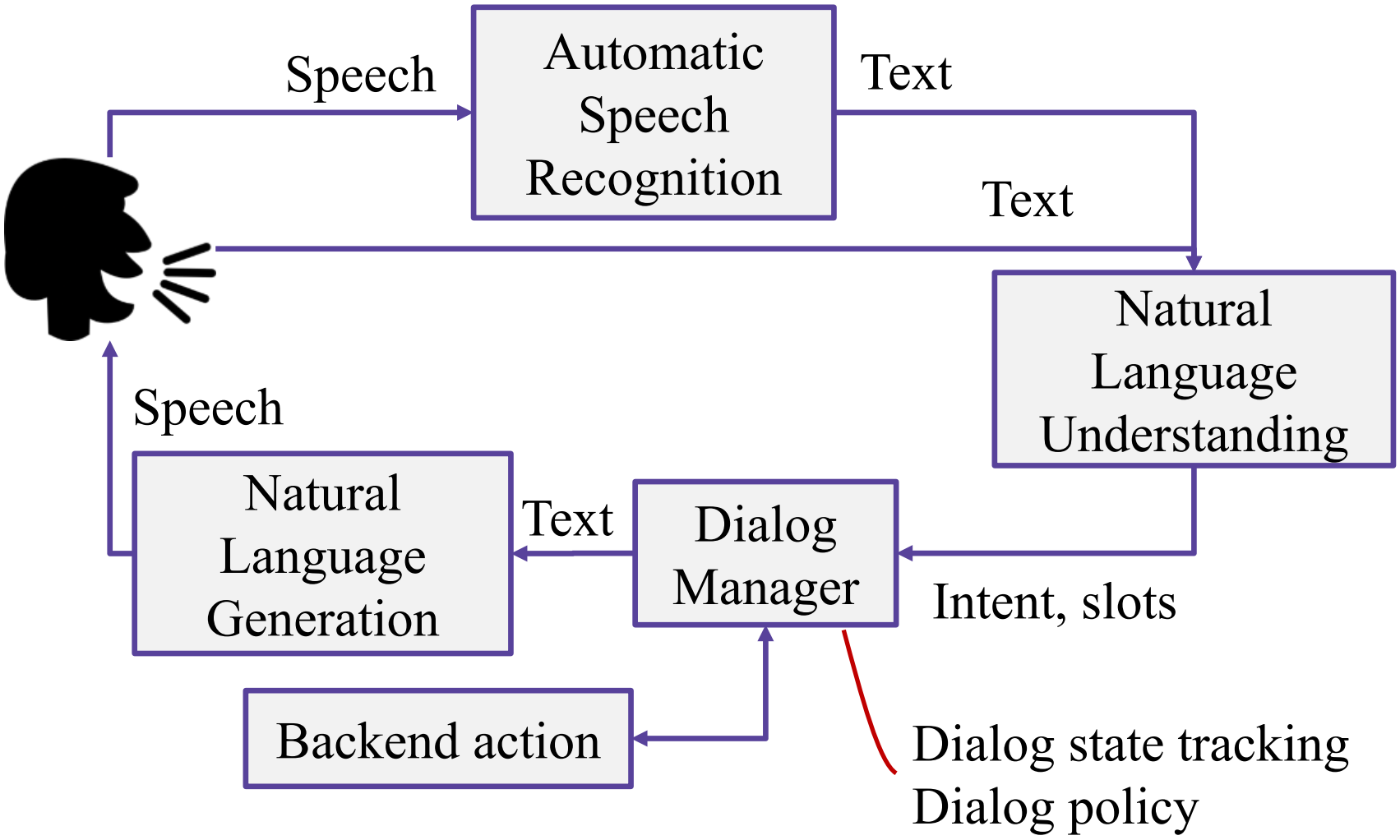


How to track a form switch

- User: **Give me directions from Los Angeles**
  - Intent classifier: nav.directions
  - Slot tagger: @FROM{Los Angeles}
  - Dialog manager: *required slot is missing, where to?*
- Agent (assistant): **Where do you want to go?**
- User: **Forget about it, let's eat some sushi first**
  - Intent classifier: nav.find
  - Slot tagger: @CATEGORY{sushi}
  - Dialog manager: *okay, let's start a new form and find some sushi*
- Agent (assistant): **Okay, here are nearby sushi places**



Task-oriented dialog system overview



Young, 2000

Summary

- We've overviewed a task-oriented dialog system with form filling
- We evaluate **accuracy** for intent classifier and **F1-measure** for slot tagger
- In the next video we'll take a closer look at intent classifier and slot tagger