CS230: Lecture 10

Class wrap-up

Kian Katanforoosh, Andrew Ng

Menti code: 21 64 9

Today's outline

I. Case Study: Conversational Assistant

II. Menti prize winners

III. Class project advice

IV. What's next?

V. Closing remarks

Open Menti question

***Goal**: Build a Chatbot to help students find and/or enroll in the right course.*

Example 1

***Student**: Hi! I want to enroll in CS106A for Winter 2019 to learn coding.*

***Chatbot**: For sure, I just enrolled you!*

Example 2

***Student**: Hi! What are the undergraduate-level History classes offered in Spring 2019?*

***Chatbot**: Here's the list of History classes offered in Spring 2019:*

. . . .

I. Case study: Conversational Assistant

Utterance

User input.

Intent

Denotes the intention of the user. Here, possibilities are enroll(), inform() etc.

Slot

Slots are used to gather multiple information from the user about their intent. Oftentimes, the user would provide some initial slots. Those should be saved so that the dialog system asks only for the details they don't have.

For the intent enroll, the slots can be "code", "quarter", "year", "SUid" etc.

Our assumption

We work in an environment with limited intents and slots.

I. Case study: Conversational Assistant

How to detect the intent?

1. Data? Input? Output?

(user utterance, intent)

x = <p> <p> Hi! I want to enroll in CS106A for Winter 2019 to learn coding. <p> <p>

y = enroll() ⟶ Call Axess

x = Hi! What are the undergraduate-level History classes offered in Spring 2019?

y = inform() ⟶ Call Explorecourses

The classes are: enroll() and inform().

Train a sequence classifier

Open Menti question

How to detect the slots?

1. **Data? Input? Output?**

(user utterance, slot tags)

x = <p> <p> show me the **Tuesday 12/05** flights from **Paris** to **Kuala Lumpur** <p> <p>

y = O O O O O **B-DAY I-DAY** O O **B-DEP** O **B-ARR I-ARR** O O

The classes are: day, departure, arrival, class, number of passengers.

x = <p> <p> Hi! I want to enroll in **CS106A** for **Winter 2019** to learn coding. <p> <p>

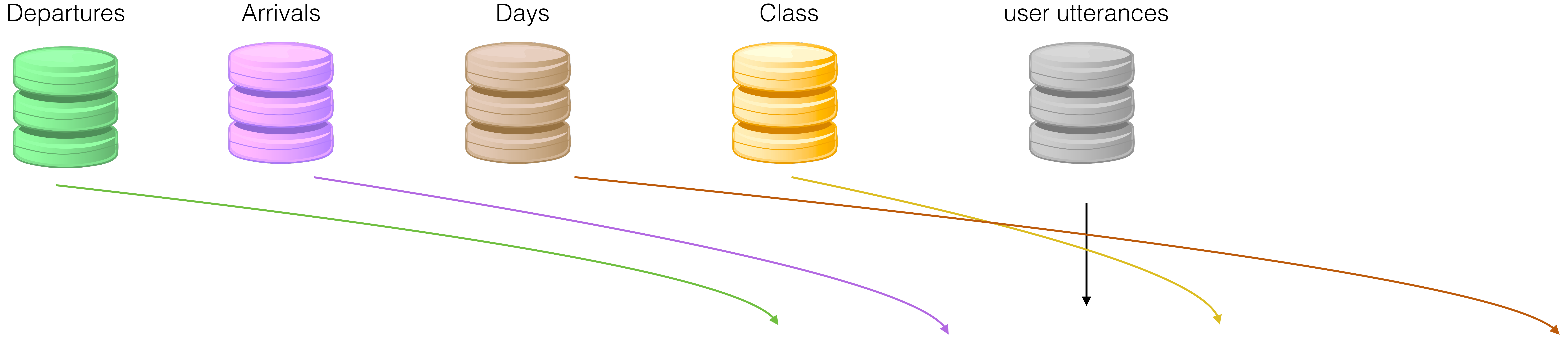
y = O O OO O O O O O **B-COD** O **B-QUA B-YEAR** O O O O O O

The classes are: code, quarter, year, SUid.

How to acquire this data?

How to detect the slots?

2. Data generation?

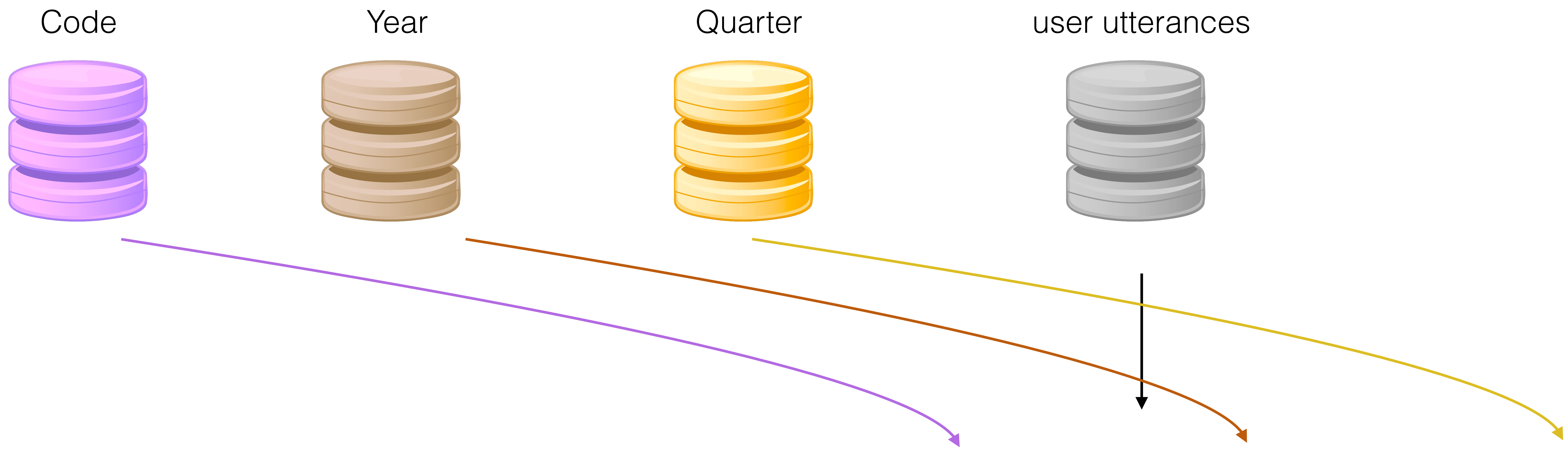


I would like to book a flight from DEP to ARR for in CLASS class for DAY.

[Bordes et al., Learning end-to-end goal-oriented dialog (2017)]

How to detect the slots?

2. Data generation?



x = Hi! I want to enroll in **CODE** for **QUARTER** **YEAR**.

y = OO O O O O O **B-COD** O **B-QUA** **B-YEAR** O

Label automatically when inserting

Train a seq2seq model

[Bordes et al., Learning end-to-end goal-oriented dialog (2017)]

I. Case study: Conversational Assistant

Example 1

Student: *Hi! I want to enroll in CS106A for Winter 2019 to learn coding.*

Chatbot: *For sure, I just enrolled you!*

Can't be understood without context.

Example 1 bis

Student: *Hi! I want to enroll in CS106A to learn coding.*

Chatbot: *For which quarter would you like to enroll?*

Student: *Winter 2019!*

Chatbot: *For sure, I just enrolled you!*

How to handle context?

I. Case study: Conversational Assistant

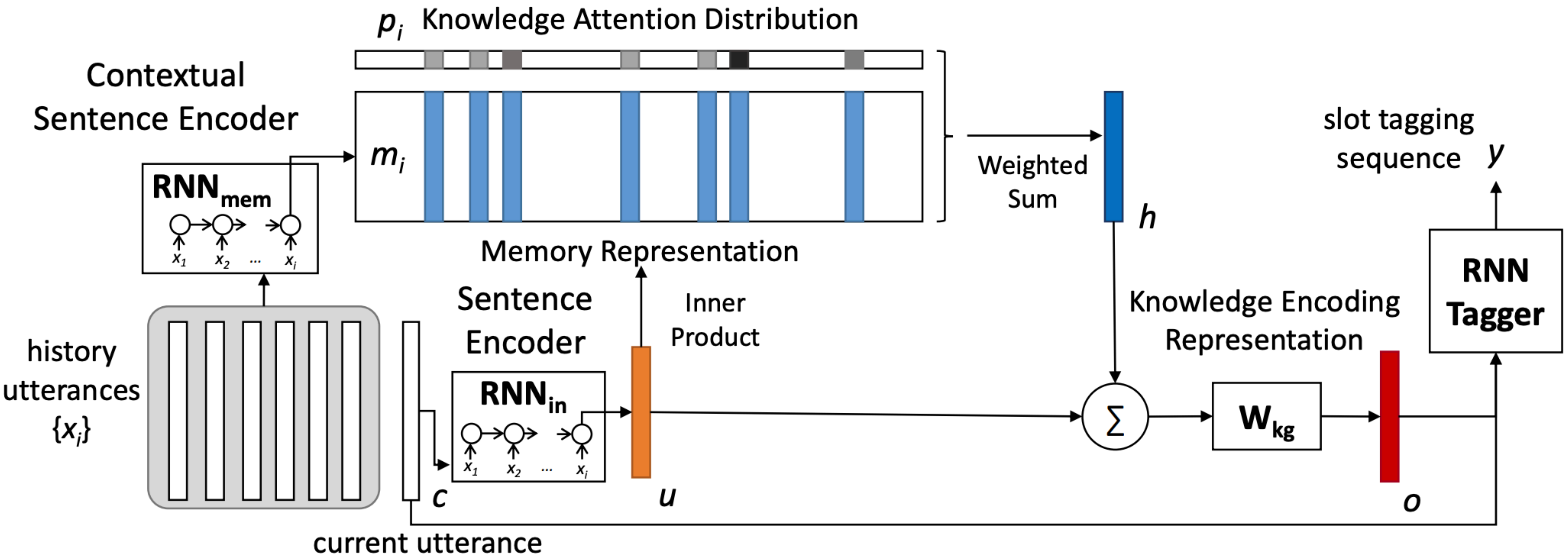


Figure 2: *The illustration of the proposed end-to-end memory network model for multi-turn SLU.*

[Chen et al., End-to-End Memory Networks with Knowledge Carryover for Multi-Turn Spoken Language Understanding (2016)]

I. Case study: Conversational Assistant

Recap'

Example

Student: *Hi! I want to enroll in a class.* Intent = enroll() Slots = quarter?year?class?

Chatbot: *Which class do you want to enroll in?*

Student: *CS230* Intent = enroll(class=CS230) Slots = quarter?year?

Chatbot: *For which quarter?*

Student: *Spring 2019!* Intent = enroll(class=CS230, quarter=spring, year=2019)

Chatbot: *You're enrolled in CS230 for Winter 2019!* Api_call = enroll(class=CS230, quarter=win_2019, SUid = ….)

How to evaluate performance?

Instructions

You will be presented with a conversation between two speakers (speaker A and speaker B).

You will also be presented with 4 potential responses from one of the speakers for this dialogue.

The task is to rate each response between 1 (inappropriate, does not make any sense) and 5 (highly appropriate and interesting) based on how appropriate the response is to continue the conversation (with 3 being neutral). A response is appropriate if it is interesting and makes sense given the previous dialogue.

If two responses are equally appropriate, you should give them the same score.

If you see a response that is not in English, please give all "1" scores.

Next

Figure 4: Instructions screen for Amazon Mechanical Turk human intelligence tasks (HITs).

Conversation	Response 1	Response 2	Response 3	Response 4
A: you need to work on your English B: Why do you say that about me? A: Well your English is very poor	But English is my native language.	What other reasons come to mind?	Here's a funny fact! Go. is the shortest complete sentence in the English language.	bye doggie
Score	4	3	3	2

Instructions:

Rate the appropriateness of the response between **1** (inappropriate, does not make any sense) and **5** (highly appropriate and interesting). The score **3** indicates neutral (acceptable, but not interesting). Remember to take into account the previous conversation.

Next

3/28

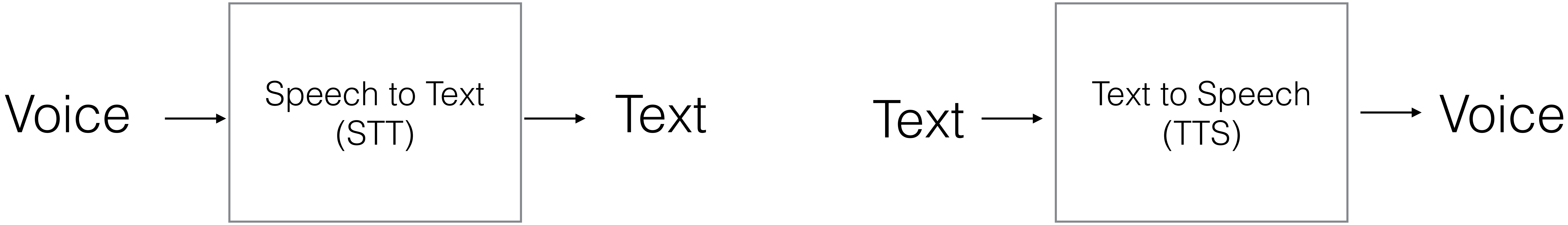
Press "Next" after filling in all the text boxes.

Figure 5: Annotation screen for Amazon Mechanical Turk human intelligence tasks (HITs). The dialogue text is a fictitious example.

[Serban et al., A Deep Reinforcement Learning Chatbot (2017)]

I. Case study: Conversational Assistant

What if you want a vocal assistant?



[Amodei et al., Deep Speech 2: End-to-End Speech Recognition in English and Mandarin (2015)]
[Shen et al., Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions (2018)]
[van den Oord et al., WaveNet: a Generative Model for Raw Audio (2016)]

Today's outline

I. Case Study: Conversational Assistant

II. Menti prize winners

III. Class project advice

IV. What's next?

V. Closing remarks

II. Menti Prize Winners

Question: computation time

Text sequence tagging is the task to output a class (or "tag") for each word of an input text sequence. For instance, the input "I want to go to Burkina Faso" can result in the following prediction: "O O O O O B-LOC I-LOC" where O indicates that the word is not a location, B-LOC (res. I-LOC) indicates that the word is the beginning (resp. inside) word of a location.

You are discussing three possible approaches with your teammates: Fully-Connected Neural Networks (FCNN), Recurrent Neural Networks (RNN) and 1-D Convolutional Neural Networks (CNN). Which of the following are true? (Check all that apply.)

a) At test time, CNN will probably be faster than the RNN because it can process the input sequence in parallel.

b) If you are using GPUs, the CNN will probably be faster than the RNN because it is optimized for GPUs.

c) If the window size of the CNN is small (let's say 3) , the FCNN will perform better than the CNN on long sequences such as "<pad> <pad> I am not sure I am available this summer, but I hope I could go to Venezuela <pad> <pad>".

d) During training, CNN will probably be faster than the RNN because it can process the input sequence in parallel.

e) None of the above.

II. Menti Prize Winners

Question: end-to-end model

You want to count the number of fish in the aquarium based on images from a camera facing the aquarium. Assume there's nothing else than fishes and that we can neglect occlusion between fishes.

You are considering using one of the two following approaches:

- **(A)** Input an image (x) to a neural network and have it directly learn a mapping to make a prediction for the number of fishes in the aquarium.
- **(B)** A two-step approach, where you would first (i) detect the fishes in the image (if any), then (ii) sum the predicted bounding boxes to get the number of fishes in the aquarium.

Between these two, Approach A seems more promising if you have a ________. (Fill-in the blank.)

a) Large training set.
b) Multi-task learning problem.
c) Large bias problem.
d) Problem with a high Bayes error.

II. Menti Prize Winners

Menti question

You collect a very large dataset and split it into a train and held-out test set. After training your deep learning model, you happily observe that it gets very high performance on both the train and test sets, and you are ready to deploy your model into the real world. Which, if any, of the following problems might still occur?

a) Because the model parameters are only local minima of the objective function, your model might not fit the training data well, which would result in poor performance on real world data.

b) High performance on the training set is indicative of overfitting, so your model might not generalize to real world data.

c) The distribution of real world data might be different from the distribution of data in your dataset, so your model might not perform well on it.

d) None of these problems will occur, because your model gets high performance on a held-out test dataset.

I. Case Study: Conversational Assistant

II. Menti prize winners

III. Class project advice

IV. What's next?

V. Closing remarks

III. Project advice

Problem description
Hyperparameters tuning & Architecture search
Paper writing
Explanations of choices and decisions (architecture, loss, metrics, data)
Data cleaning and preprocessing (if applicable)
How much code you wrote on your own
Insights and discussions (including next steps, and interpretation of results)
Results: Accuracy (or other metric) satisfaction
References
Penalty for more than 5 pages (except References/ contribution/theory-proofs)

For the final poster presentation:

- You will have 3min to pitch.
- Followed by ≈2min of questions

Today's outline

I. Case Study: Conversational Assistant

II. Menti prize winners

III. Class project advice

IV. What's next?

V. Closing remarks

Classes at Stanford

Natural Language Processing

CS 124: From Languages to Information (LINGUIST 180, LINGUIST 280)
CS 224N: Natural Language Processing with Deep Learning (LINGUIST 284)
CS 224U: Natural Language Understanding (LINGUIST 188, LINGUIST 288)
CS 276: Information Retrieval and Web Search (LINGUIST 286)

Computer Vision

CS 131: Computer Vision: Foundations and Applications
CS 205L: Continuous Mathematical Methods with an Emphasis on Machine Learning
CS 231N: Convolutional Neural Networks for Visual Recognition
CS 348K: Visual Computing Systems

Others:

CS 273B: Deep Learning in Genomics and Biomedicine (BIODS 237, BIOMEDIN 273B, GENE 236)
CS 236: Deep Generative Models
CS 228: Probabilistic Graphical Models: Principles and Techniques
CS 337: AI-Assisted Care (MED 277)
CS 229: Machine Learning (STATS 229)
CS 229A: Applied Machine Learning

CS 234: Reinforcement Learning
AI for Healthcare Bootcamp

Today’s outline

I. Case Study: Conversational Assistant

II. Menti prize winners

III. Class project advice

IV. What’s next?

V. Closing remarks

Announcements

Finals week schedule

- TA Sections (Project advice): Friday 12/7
- Poster Session: Friday 12/14
- Final Project Report Due: Sunday 12/16, 11:59pm