Hybrid Model for Goal Oriented Dialogue Generation

Ana Valeria González
University of Copenhagen

But first a quick overview of dialogue systems…

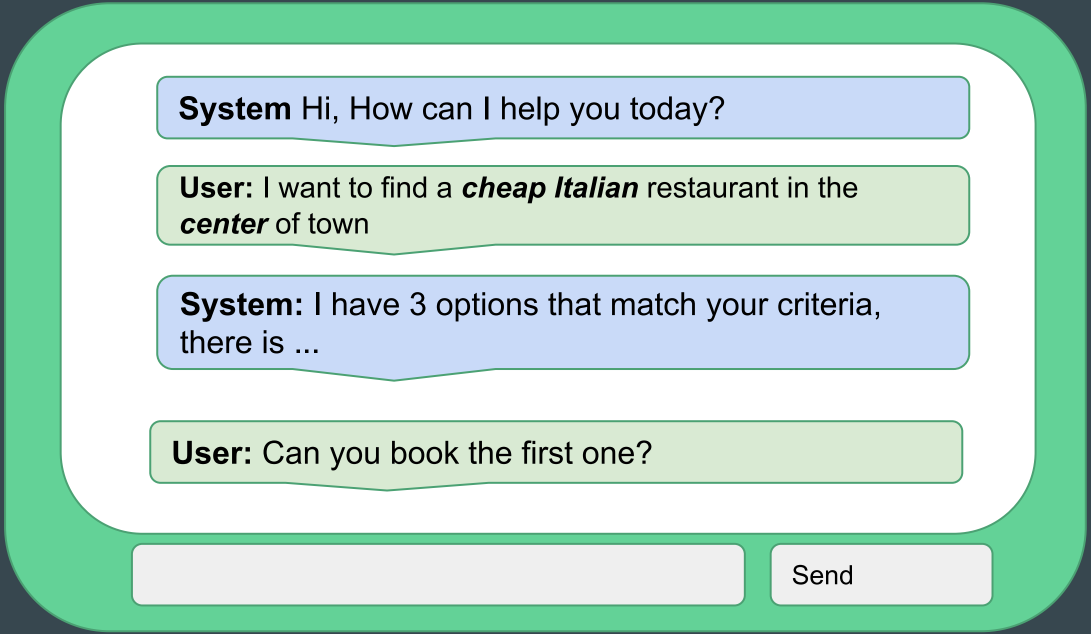
Dialogue Systems

Much of the research separates Dialogue into two subfields:

- **Open ended dialogue generation**: typically using encoder-decoder architectures
- **Goal Oriented dialogue systems**: Some work using encoder-decoders, however, much more using pipeline methods and state tracking.

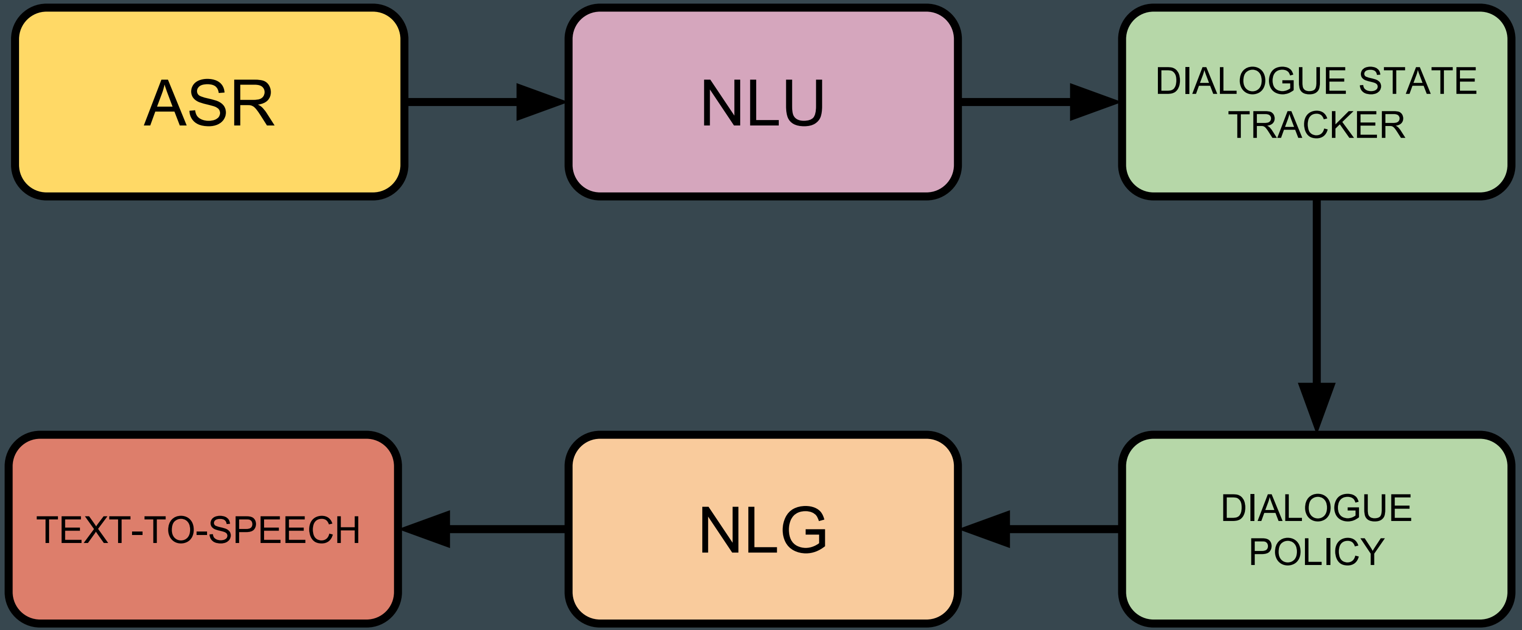
Goal Oriented Dialogue

There is an end goal to be achieved by the system and determined by the user. Taking an example for customer support

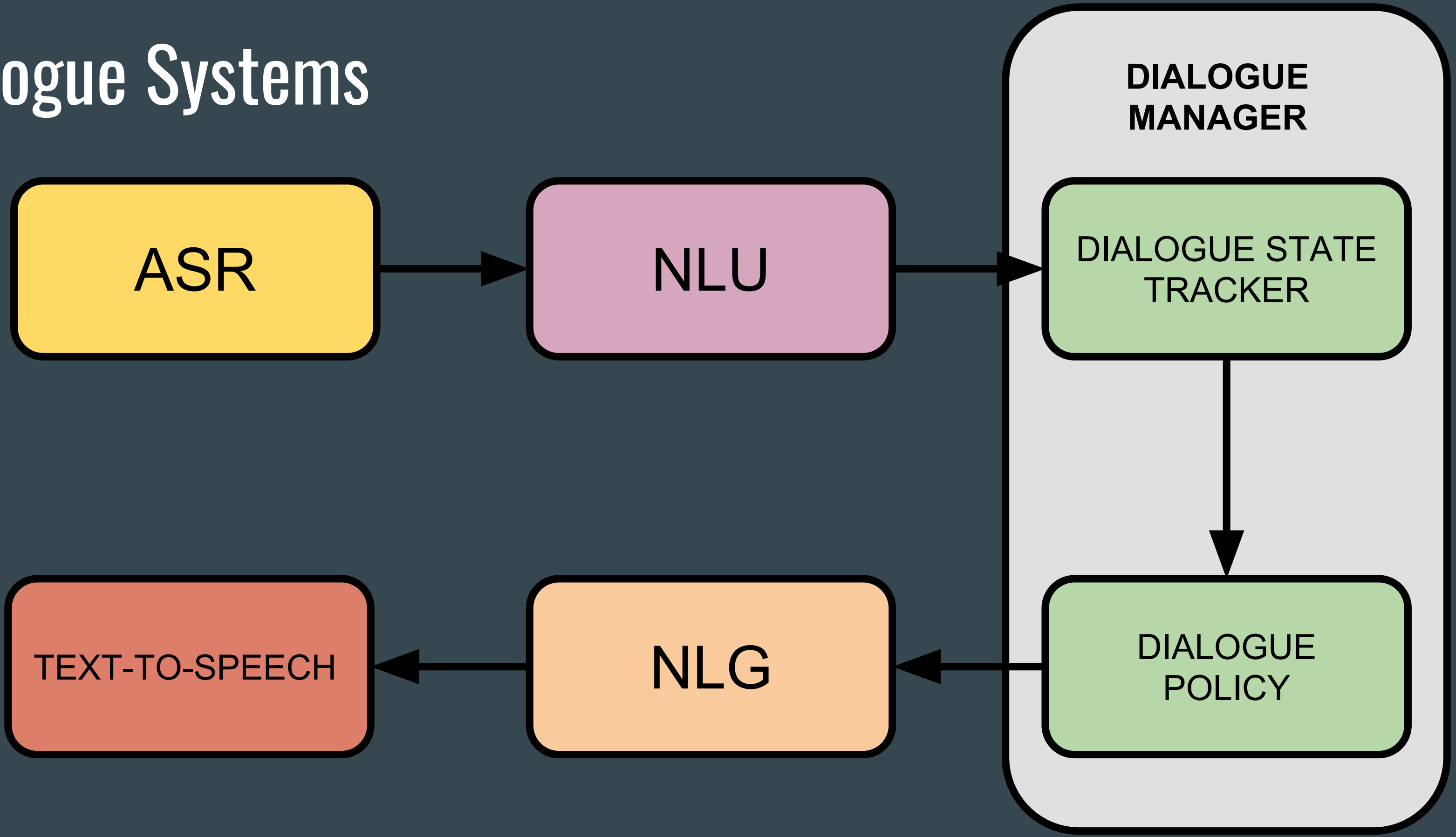


A standard pipeline...

Dialogue Systems



Dialogue Systems



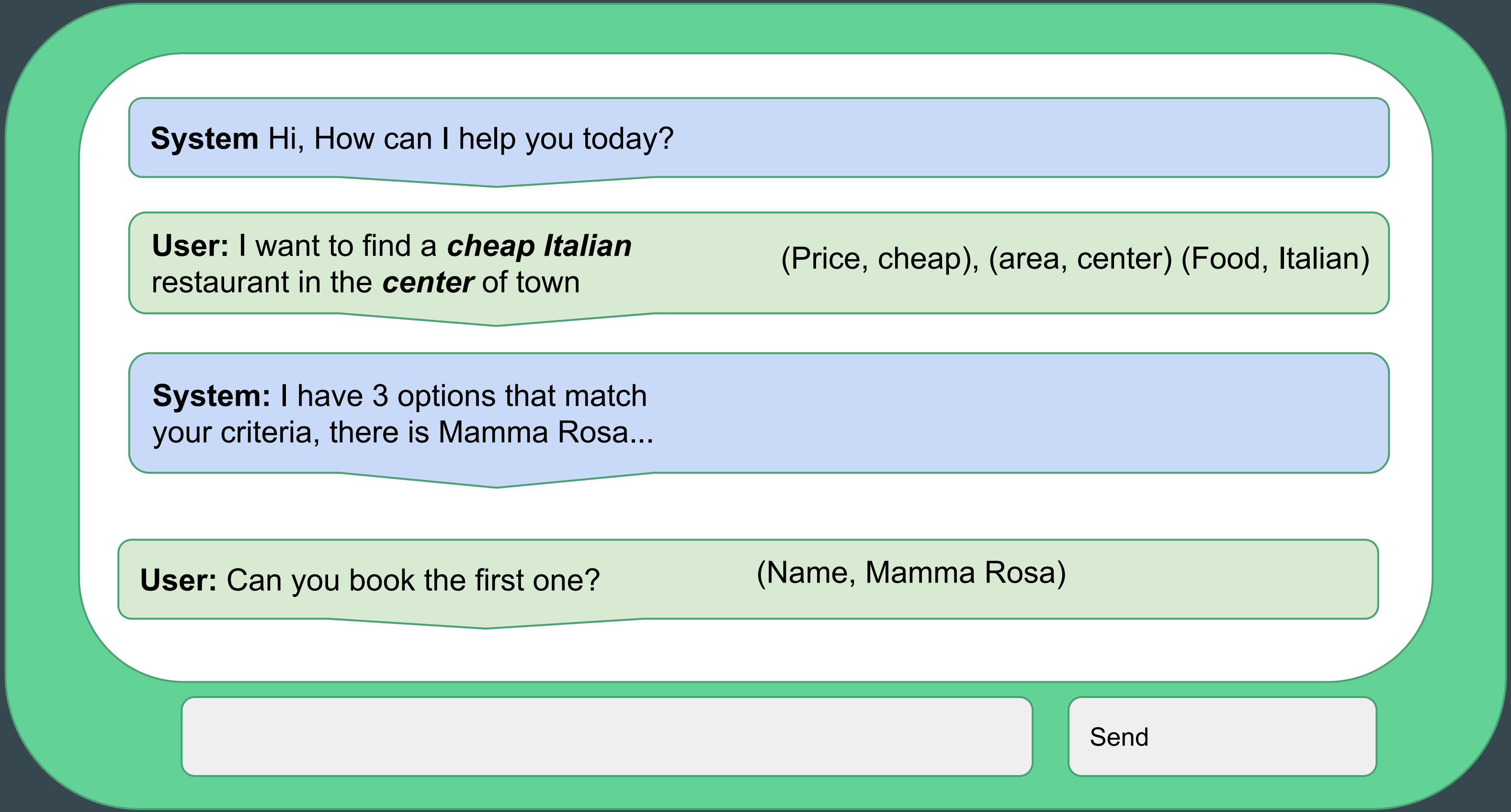
Dialogue Management

- **Dialogue State Tracker:** Detecting the user intent and goals
- **Dialogue Policy:** Takes the stated detected by the DST module and decides what action to take next

Dialogue State Tracking

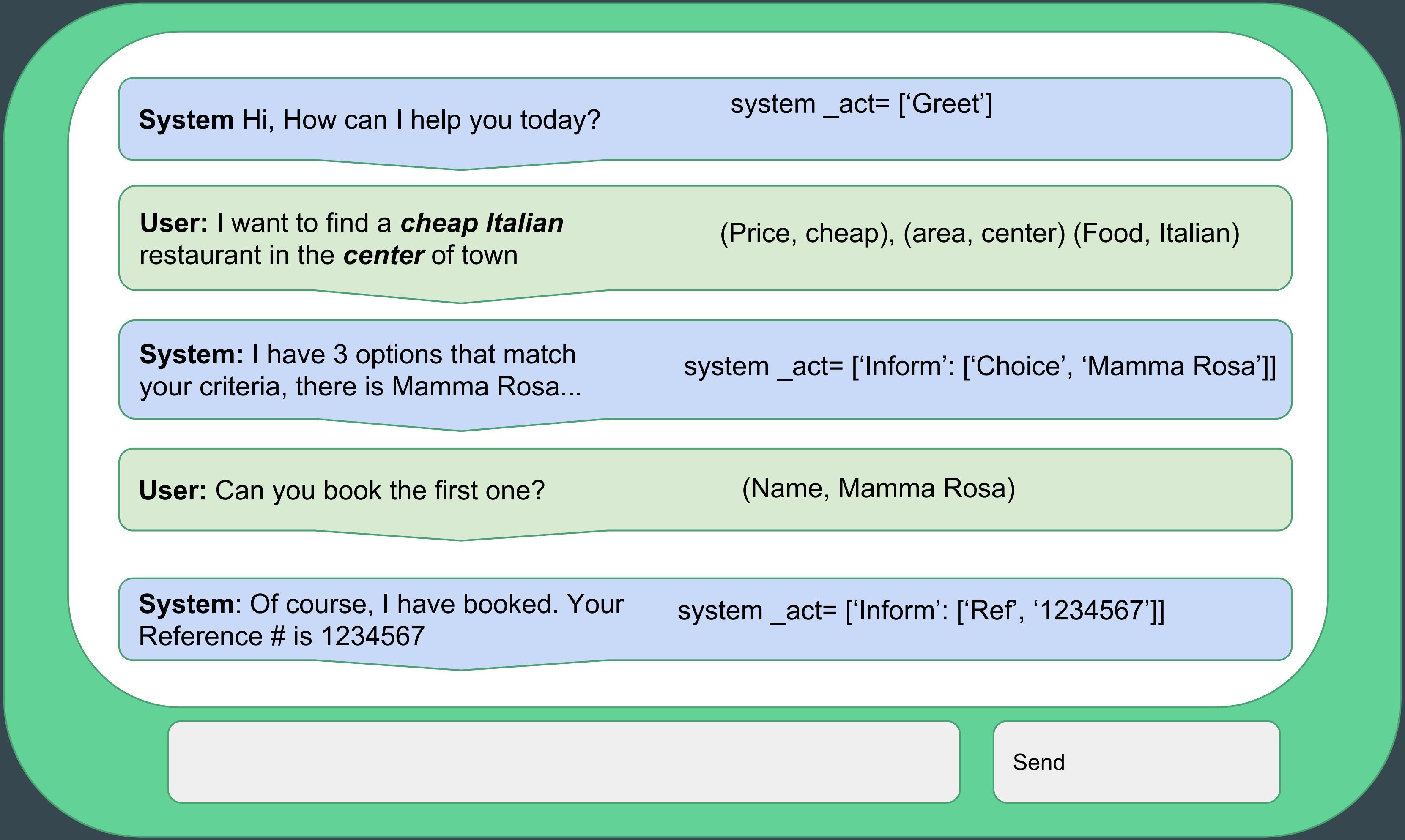
- Inferring the ***user intent*** or the belief state at a given turn in the conversation
- Intents are typically defined by a **domain ontology**
  - All possible user intents (slots and values) that the system can handle are specified here

Dialogue State Tracking



Dialogue Policy

- Using the detected user intent to make a decision about what action to take next


System Hi, How can I help you today?
system _act= ['Greet']
User: I want to find a cheap Italian restaurant in the center of town
(Price, cheap), (area, center) (Food, Italian)
System: I have 3 options that match your criteria, there is Mamma Rosa...
system _act= ['Inform': ['Choice', 'Mamma Rosa']]
User: Can you book the first one?
(Name, Mamma Rosa)
System: Of course, I have booked. Your Reference # is 1234567
system _act= ['Inform': ['Ref', '1234567']]
Send

But there are some issues...

- Slot, value pairs are typically ***fixed.***
    - Hard to adapt models to other domains or new slot values
- Recently introduced end to end approaches for state tracking
    - The embedding-based representation of slots and values makes it a natural choice for a domain transfer.

More on Goal Oriented Dialogue

Outside the scope of customer support, this can also include:

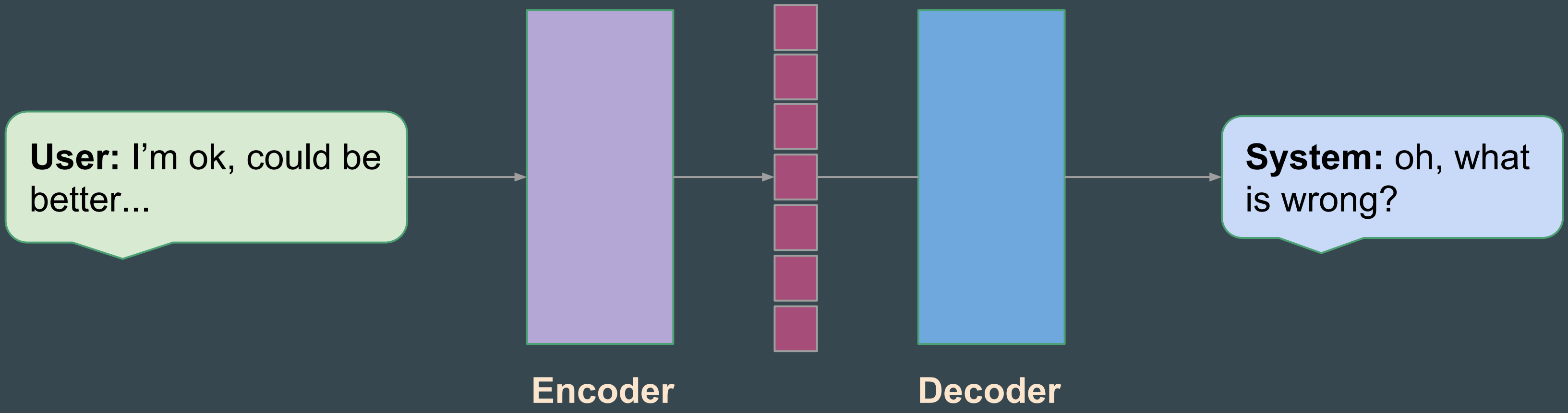
- Tutoring systems
- Personal Assistants i.e. Cortana, Google Assistant, Siri, Alexa

Goal oriented systems incorporate ***Information Retrieval*** **,** ***Question Answering and Machine Reading*** methods

Open-ended Dialogue Generation

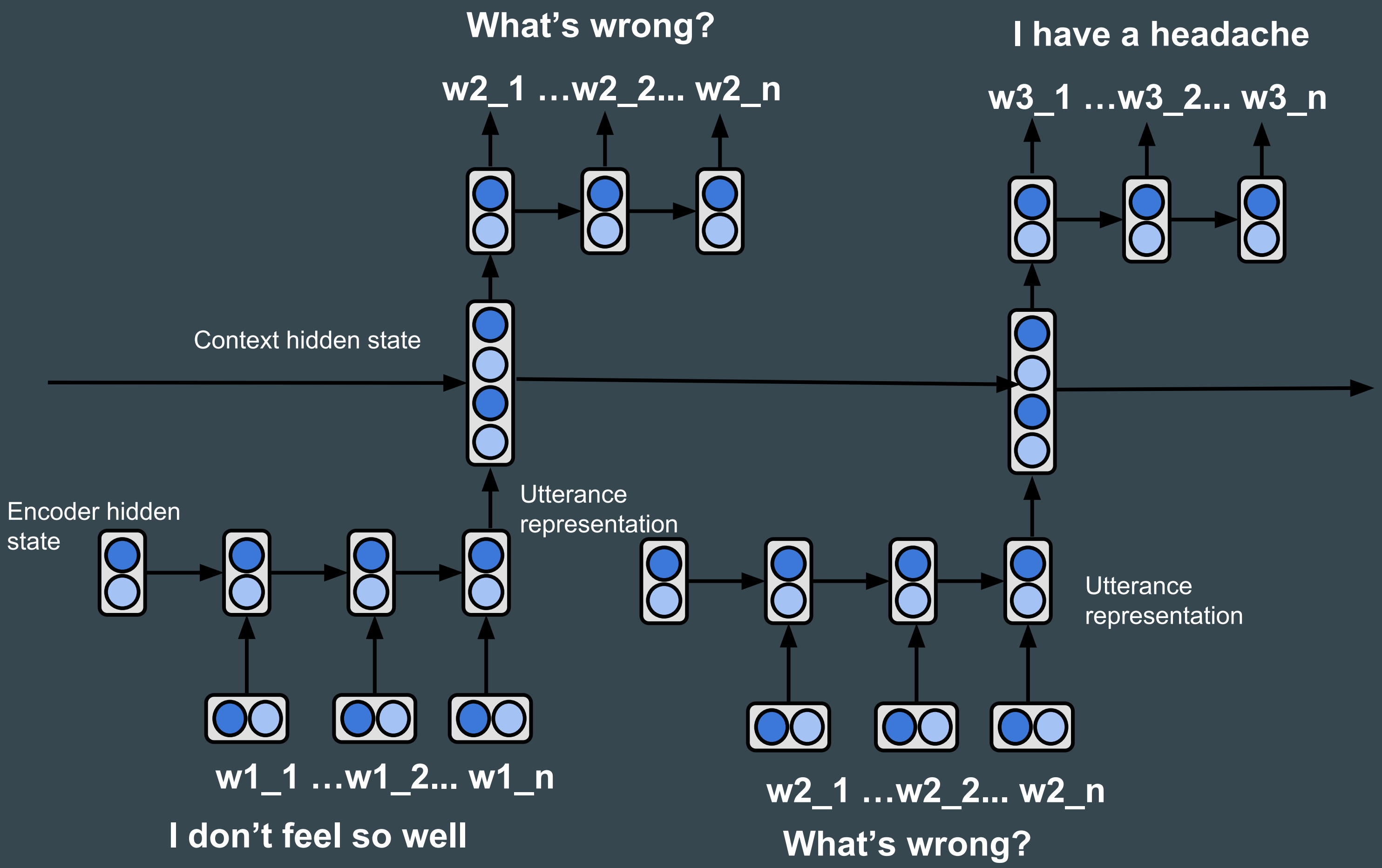
Encoder-Decoder models

- Dialogue as a machine translation problem



Hierarchical encoder-decoder models

- **Encoder RNN**
- **Context RNN:** Keeps track of past utterances by processing iteratively each past utterance vector
- **Decoder RNN:** Conditioned on the hidden states of the Context RNN


What's wrong?
w2_1 …w2_2... w2_n
I have a headache
w3_1 …w3_2... w3_n
Context hidden state
Utterance representation
Encoder hidden state
Utterance representation
w1_1 …w1_2... w1_n
I don't feel so well
w2_1 …w2_2... w2_n
What's wrong?

But of course these create fluent but (often) meaningless dialogues

User: hi

System: hey! How are you?

User: i’m ok you?

System: I don’t know

User: Why don’t you know?

System: I don’t know

User: mmm that is odd

System: I don’t know

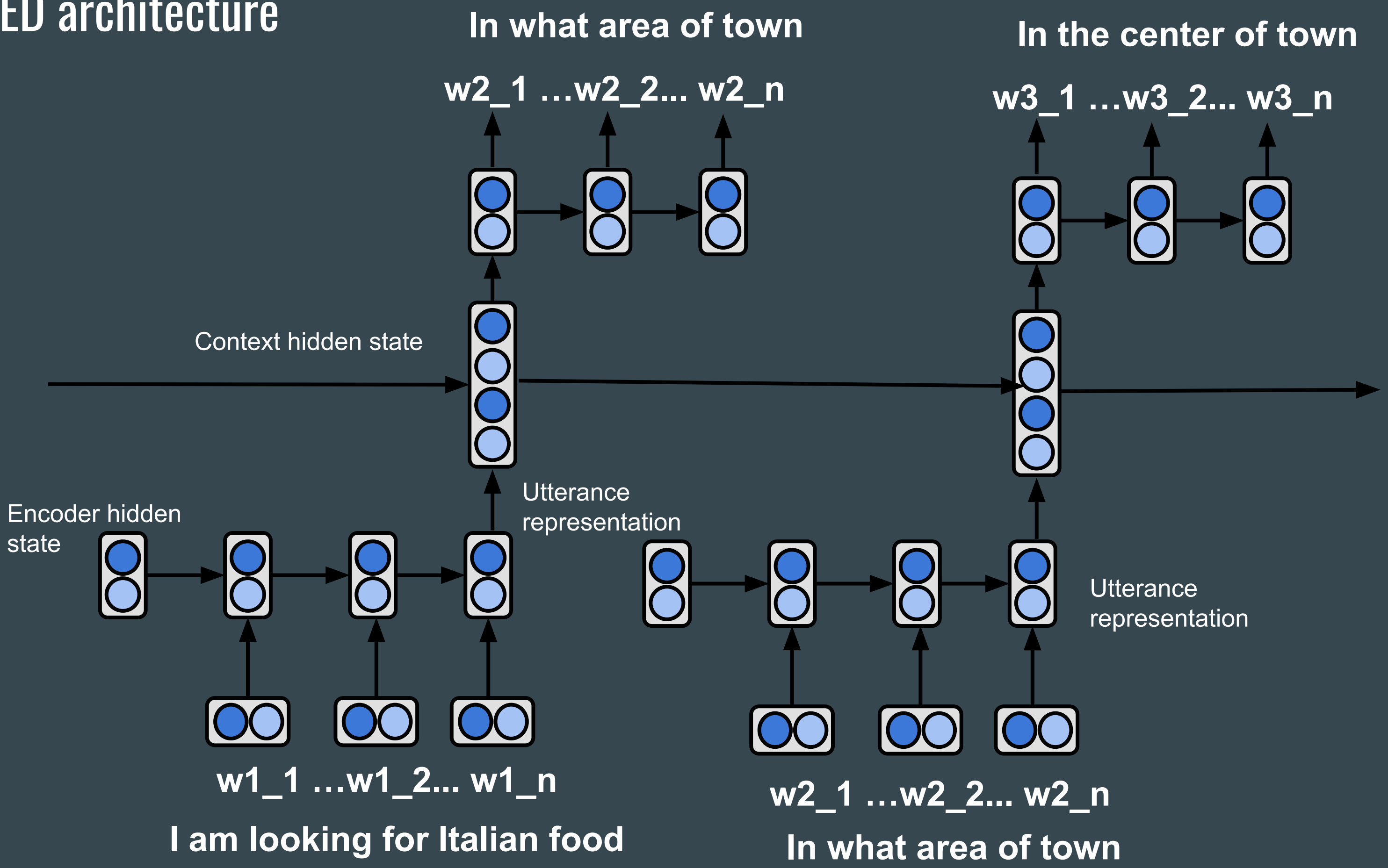
Hybrid Model for Goal Oriented Dialogue Generation

Preliminary results

Bridging the gap

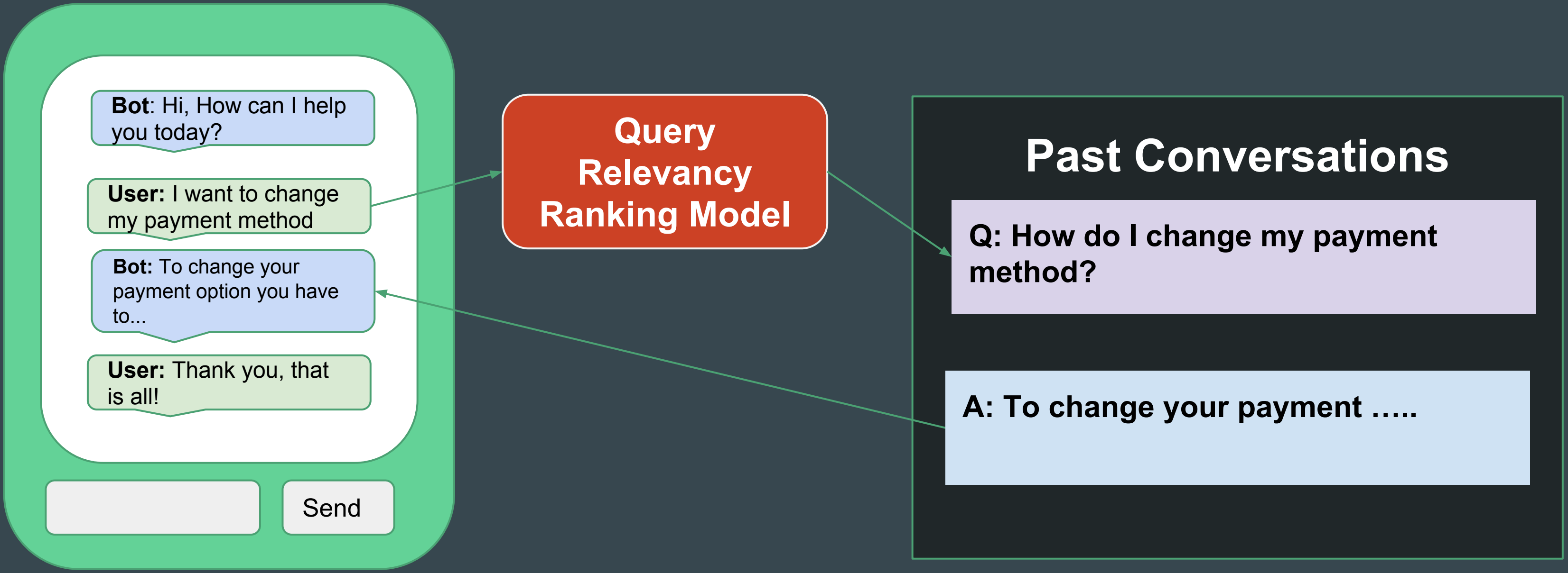
- Hierarchical methods useful for ***open ended dialogue.***
- ***Goal-directed dialogue*** has focused almost exclusively on dialogue state tracking
- dialogue generation for goal-directed dialogue

HRED architecture



Problem Definition

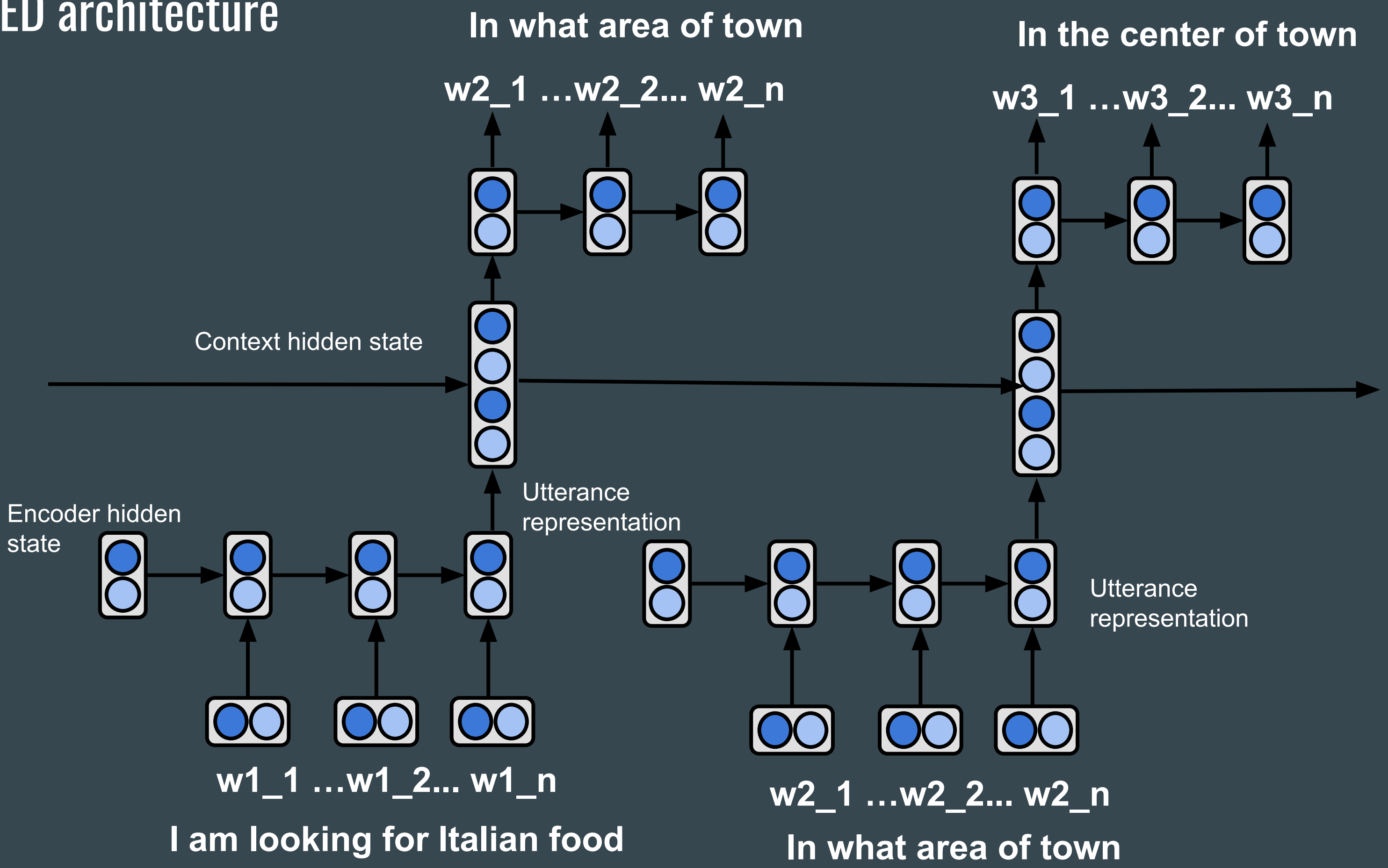
- Given a new user enquiry, can we use past conversations to inform our dialogue generation model

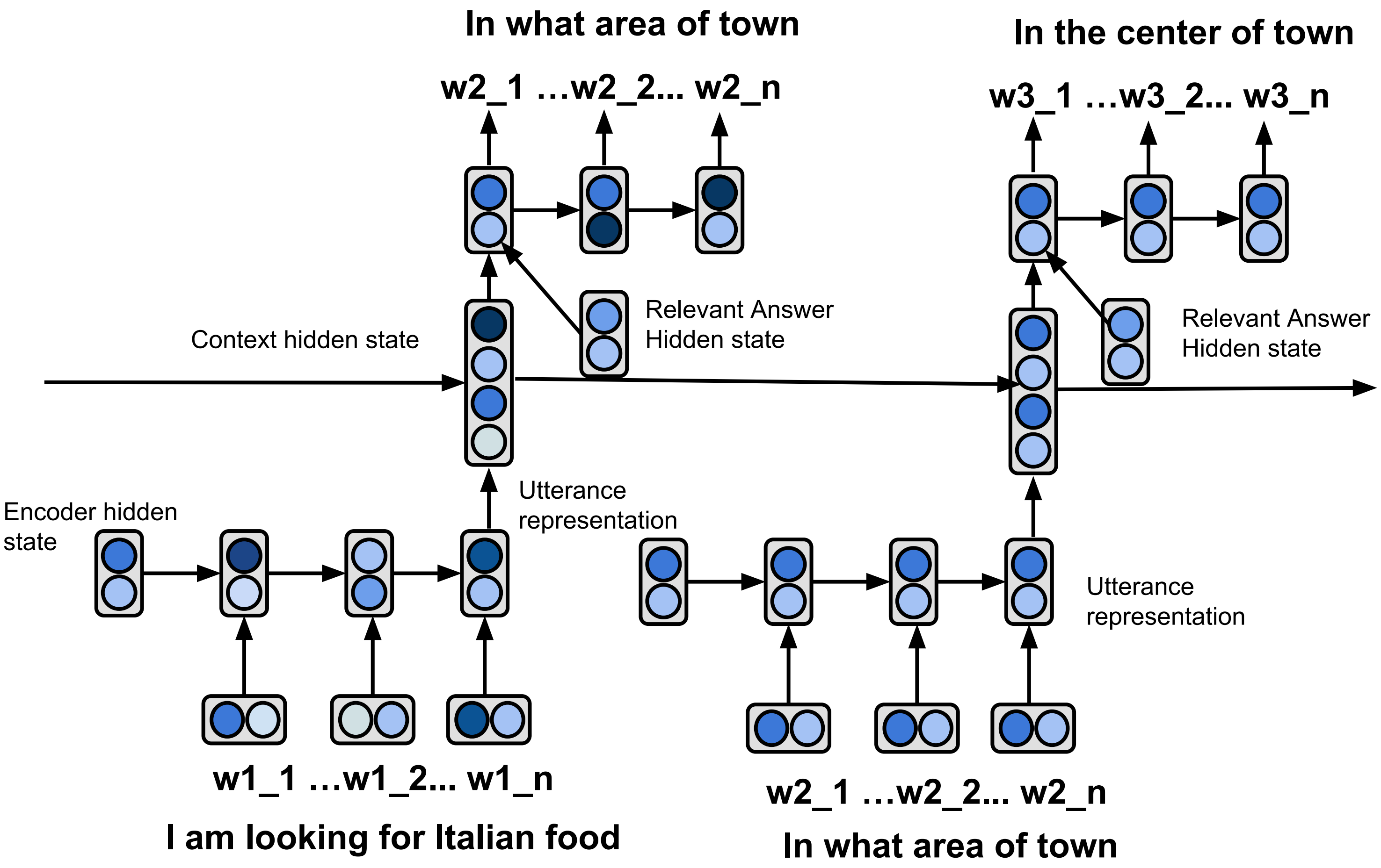


Past Conversation Retrieval

- Given a new query, we retrieve the top ten related queries from the training set
  - Approximate nearest neighbors
- Rerank using a feed forward ranker by **González et al. 2018**
- Use the answer to the most highly ranked related query to condition our decoder

HRED architecture





In what area of town
w2_1 …w2_2... w2_n
In the center of town
w3_1 …w3_2... w3_n
Relevant Answer
Hidden state
Context hidden state
Relevant Answer
Hidden state
Utterance
representation
Encoder hidden
state
Utterance
representation
w1_1 …w1_2... w1_n
I am looking for Italian food
w2_1 …w2_2... w2_n
In what area of town

MultiWOZ dataset

Dialogues	8,430
Total turns	115,424
Total tokens	1,520,970
Avg. turns per dialogue	13.68
Avg. tokens per turn	13.18
Total unique tokens	24,071

Some preliminary results

	HRED	HYBRID
VECTOR EXTREMA	52.5	54.1
AVG EMB. SIMILARITY	90.0	91.0
GREEDY MATCHING	22.7	22.9

What do these metrics mean?

Dialogue Success prediction

- Are our generated responses implicitly learning the belief state of the conversation at a given point?
  - Trained a simple model to detect dialogue acts
  - Compared performance on our system outputs versus the ground truth

What do these metrics mean?

Dialogue Success prediction

- Are our generated responses implicitly learning the belief state of the conversation at a given point?

Model	Accuracy
MultiWOZ benchmark	60.29
LSTM-System transcript	62.76
LSTM-Baseline Output	33.02
LSTM-Hybrid Output	39.59

Acts	**Hybrid**	**HRED**	**Gold**
INFORM: 'Id', 'Leave'	I have a train that leaves at .	Booking was successful, the total fee is is GBP	TR4276 leaves at 15:59, will this option work for you?
INFORM: 'Choice', 'Day'	There are 133 trains that meet your criteria. What day ?	What day and time you would you like to ?	There are 133 trains traveling that path. What day would you like to travel on?
INFORM: 'Addr'	It is located at Hills Rd, Cambridge. Is there anything else I can help you with?	Sure. The address is street and	Doubletree by Hilton Cambridge Granta Place Mill Lane is the address

Considerations

- This approach is a start towards bridging the gap however it is still *naive*
- Integrating **logic, anaphora resolution** with **retrieval methods**
- This assumes that our new queries (or similar enough queries) have been asked before.... Not always the case

How can we tackle ***domain adaptation*** in goal oriented dialogue?

Domain Transfer for DST using Reinforcement Learning

On going work...

Domain Transfer

- Turn level supervision to learn a model for domains with turn level annotations
- finetune to a domain where turn level labels are not available
- Modelling dialogue level user feedback in a new domain and using it as a reward

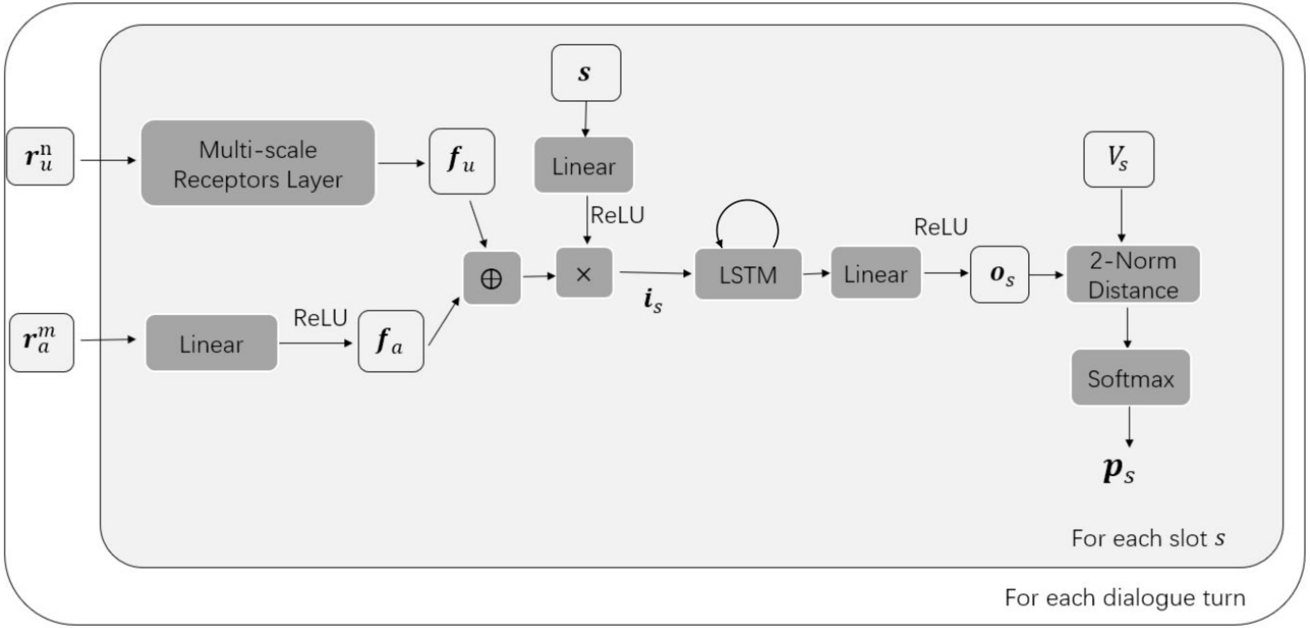
StateNET DST model

- 1.Representing Utterances
  - Uses pretrained embeddings and doesn't train them
  - C number of linear layers take representation of as input
  - ReLU activation followed by linear layer to create a feature vector
- 2.Representing Slots
  - Linear layer
- 3. Representing previous acts
  - Linear Layer

Concatenation of 1 and 3 , multiplied with 2, fed into LSTM

---> Fixed length prediction, compared word vectors of slots value

StateNet DST model

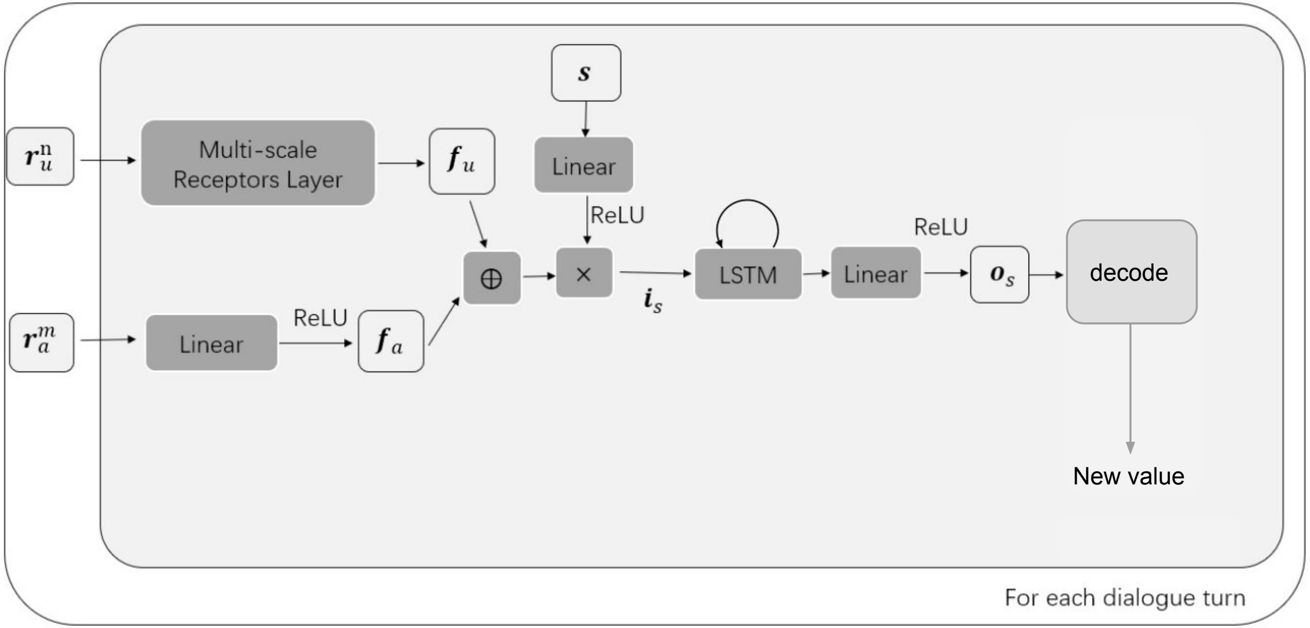


Pretraining

- Using N out of M domains with turn level supervision (slot,value labels) learn a model the same way as StateNET

Domain Transfer using RL

- For a domain not included in the pretraining domains, we use dialogue level goal accuracy to model user satisfaction.
    - Assumption: getting something like 80% accuracy would translate to ⅘ stars user rating
- Using “user satisfaction” as our reward to fine tune our model to new domains
- Using vector representations of slot value rather than treating it as a predefined class we can decode


r_u^n
Multi-scale Receptors Layer
f_u
s
Linear
ReLU
$\oplus$
$\times$
i_s
LSTM
Linear
ReLU
o_s
decode
New value
r_a^m
Linear
ReLU
f_a
For each dialogue turn

Reward

We compute the accuracy over the predicted slot values and the ground truth for each turn

- Use the joint accuracy at the end of the dialogue to model our user satisfaction reward

Preliminary observations

- For long dialogues, it becomes very difficult to learn with this delayed reward
  - One option could be curriculum learning
  - Difficult to know which turns are more important
  -
- With 7 domains and about 40 slots (compared to the usual 1 domain and 3-4 slots), most well performing DST models become extremely inefficient

To be continued...

Questions !!