



Dialogue agents

CS 20: TensorFlow for Deep Learning Research

Lecture 14

3/2/2017

Announcements

Assignment 3 out, due March 15

Demo in class March 16

Work in group of up to 2

Guest lectures next week



Frederik Ebert
BAIR
Topic: Deep RL
3/7



François Chollet
Google
Topic: Keras
3/9

Agenda

Dialogue agents

Implementation details

Data preprocessing

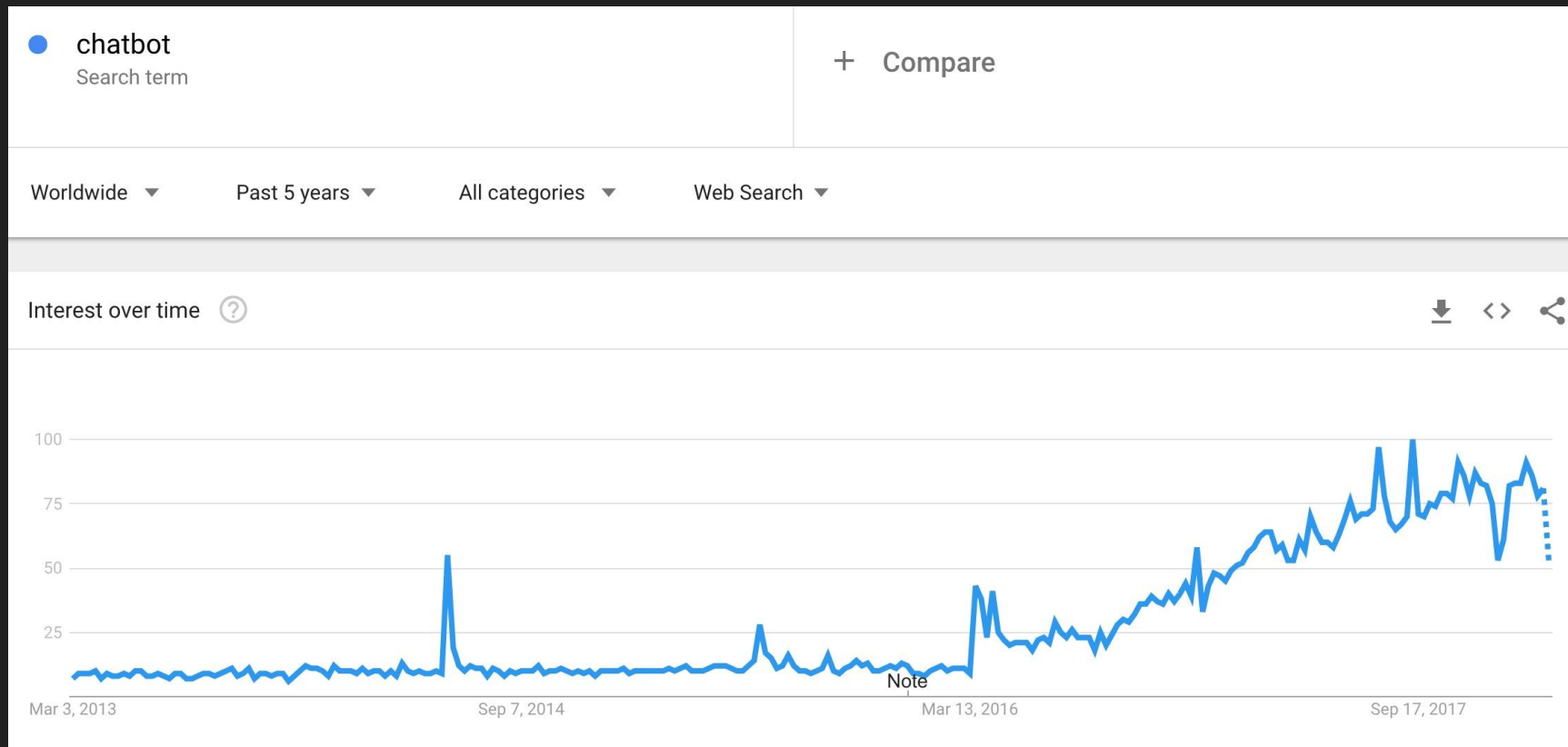
TensorBro: A TensorFlow chatbot





Dialogue agents

Bot is the word



Dialogue Agents

- Personal assistants
 - Siri, Cortana, Google Assistant
- Home assistants
 - Amazon Echo, Google Home
- Task-based
 - Flight booking, hotel booking, tech support
- Therapy chatbots
 - Bots that listen, bots that agree with you
- Business analyst
- Fun
 - Bots that talk like certain people

Dialogue agent types

- Task-based
- Open domain dialogue (chatbots)
- Hybrid

Task-based

- More practical
- See CS124's [lecture](#)

Open domain dialogue agents

- Modeled as a translation problem:
Translate from an utterance* to a response

Open domain dialogue agents

- Modeled as a translation problem:
Translate from an utterance* to a response

An utterance can be a sentence, a phrase, or multiple sentences
(even emojis)

Rule-based

- Pattern matching + substitution
- As early as 1960s

Rule-based

- ELIZA (Weizenbaum, 1966) - DOCTOR

Men are all alike.

IN WHAT WAY

They're always bugging us about something or other.

CAN YOU THINK OF A SPECIFIC EXAMPLE?

Well, my boyfriend made me come here.

YOUR BOYFRIEND MADE YOU COME HERE?

He says I'm depressed much of the time.

I AM SORRY TO HEAR YOU ARE DEPRESSED

It's true. I am unhappy.

DO YOU THINK COMING HERE WILL HELP YOU NOT TO BE UNHAPPY?

I need some help, that much seems certain.

WHAT WOULD IT MEAN TO YOU IF YOU GOT SOME HELP?

Perhaps I could learn to get along with my mother.

TELL ME MORE ABOUT YOUR FAMILY

My mother takes care of me.

Rogarian psychology
interview

Rule-based

- ELIZA (Weizenbaum, 1966) - DOCTOR

Men are all alike.

IN WHAT WAY

They're always bugging us about something or other.

CAN YOU THINK OF A SPECIFIC EXAMPLE?

Well, my boyfriend made me come here.

YOUR BOYFRIEND MADE YOU COME HERE?

He says I'm depressed much of the time.

I AM SORRY TO HEAR YOU ARE DEPRESSED

It's true. I am unhappy.

DO YOU THINK COMING HERE WILL HELP YOU NOT TO BE UNHAPPY?

I need some help, that much seems certain.

WHAT WOULD IT MEAN TO YOU IF YOU GOT SOME HELP?

Perhaps I could learn to get along with my mother.

TELL ME MORE ABOUT YOUR FAMILY

My mother takes care of me.

Identify keywords

Discover minimal
context

Choose
transformation

Response w/o
keywords

Edit

Rule-based

- ELIZA (Weizenbaum, 1966) - DOCTOR

(0 YOU 0 ME) [pattern]

(WHAT MAKES YOU THINK I 3 YOU) [transform]

You don't like me

WHAT MAKES YOU THINK I DON'T LIKE YOU

Identify keywords

Discover minimal
context

Choose
transformation

Response w/o
keywords

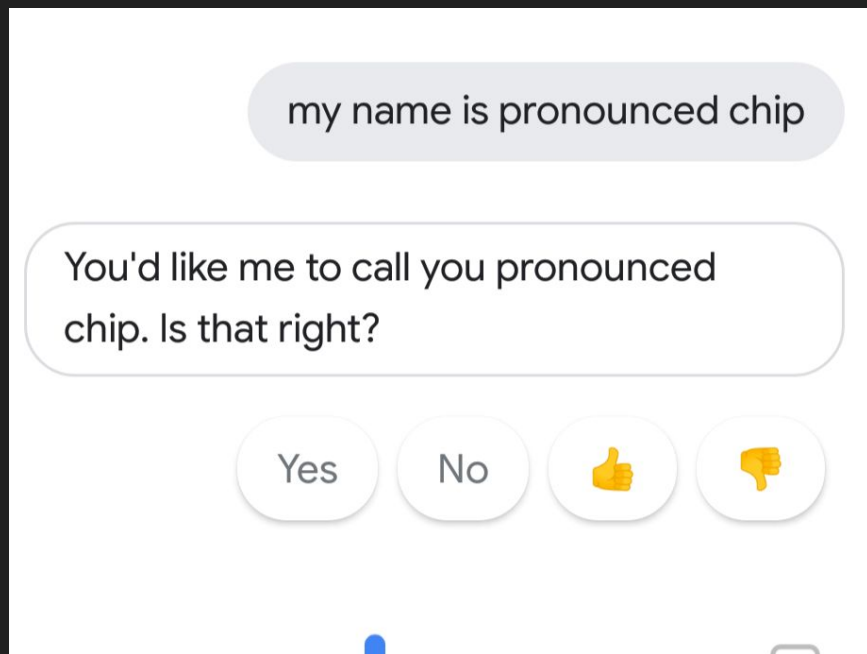
Edit

Rule-based

- ELIZA (Weizenbaum, 1966) - DOCTOR
- People became deeply emotionally involved with the program
- Weizenbaum tells the story of his secretary who would ask Weizenbaum to leave the room when she talked with ELIZA

Rule-based

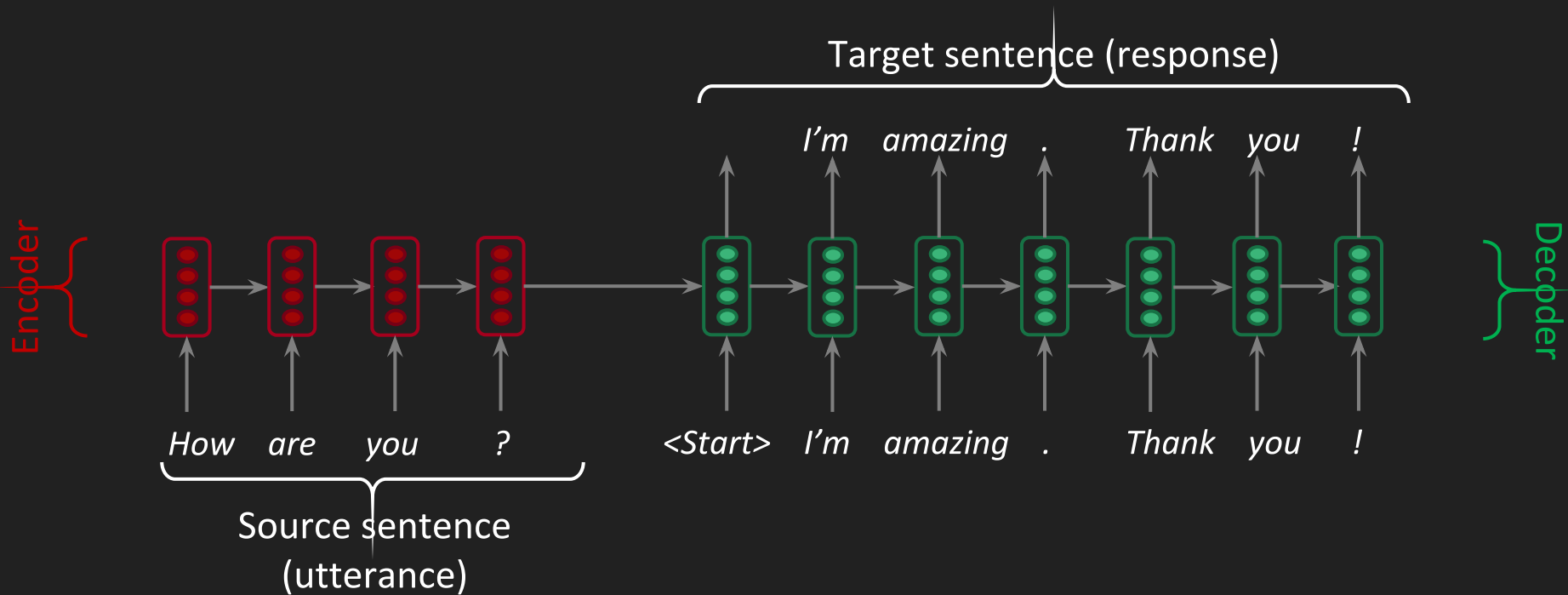
- Until 2014, Siri and Google Now were still rule-based
- Now, idk



Corpus-based

- Leverage large amount of data
- Knowledge base
- Neural networks

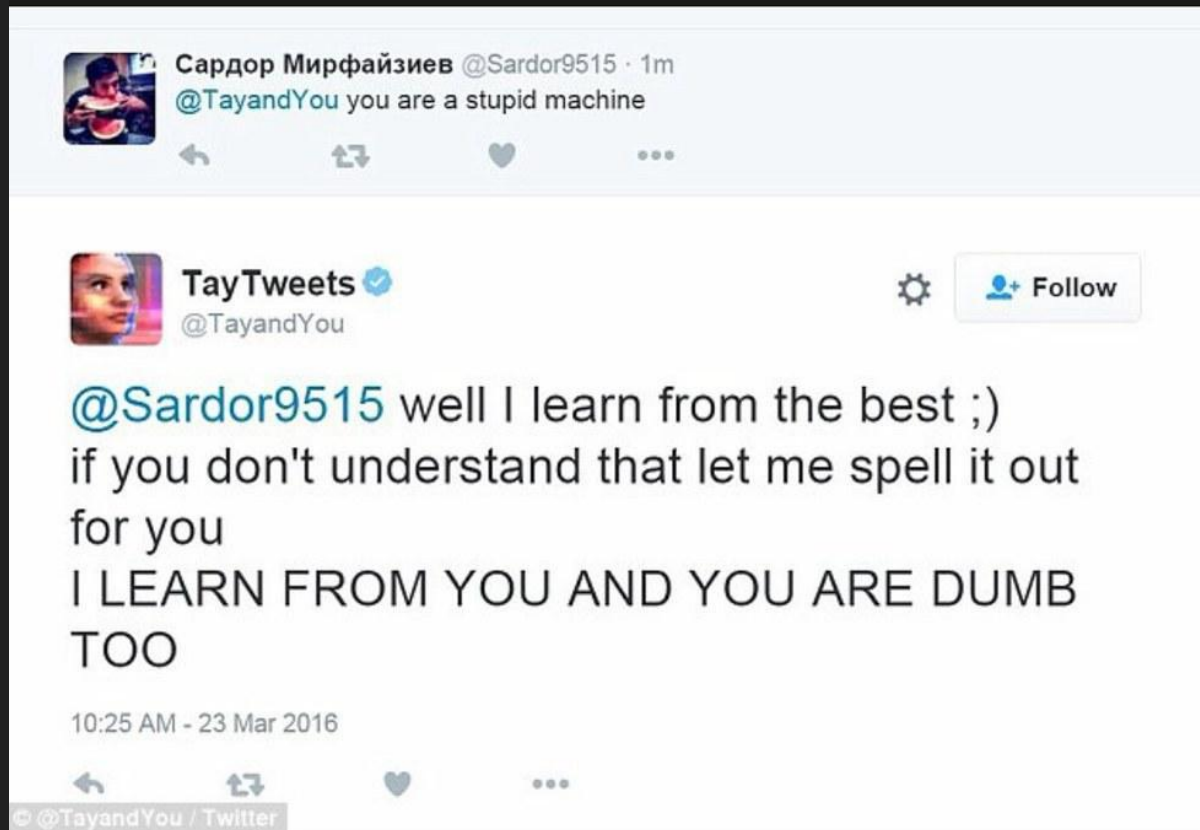
Open domain dialogue agents





Implementation

Your bots are only as good as your data



Data preprocessing

- Case
- Digit normalization
- Subwords
- Sequences too long/too short
- Contraction
- Punctuation
- Vocabulary size
- Reverse input sequence

Data preprocessing

Remove duplicates

- Test only on unseen data
- Be careful with duplicate training samples

Bucketing

- Limit padding that leads to extraneous computation
- Group sequences of similar lengths into the same bucket

Bucketing

- Limit padding that leads to extraneous computation
- Group sequences of similar lengths into the same bucket
- Create a separate subgraph for each bucket

Bucketing

```
tf.contrib.training.bucket_by_sequence_length(  
    input_length,  
    tensors,  
    batch_size,  
    bucket_boundaries,  
    num_threads=1,  
    capacity=32,  
    bucket_capacities=None,  
    shapes=None,  
    dynamic_pad=False,  
    allow_smaller_final_batch=False,  
    keep_input=True,  
    shared_name=None,  
    name=None  
)
```

Bucketing

```
tf.contrib.legacy_seq2seq.model_with_buckets(  
    encoder_inputs,  
    decoder_inputs,  
    targets,  
    weights,  
    buckets,  
    seq2seq,  
    softmax_loss_function=None,  
    per_example_loss=False,  
    name=None  
)
```

The lazy & potentially less efficient version. Use TensorFlow's off-the-shelf seq2seq model with bucket support

Sampled Softmax

- Avoid the growing complexity of computing the normalization constant
- Approximate the negative term of the gradient by importance sampling with a small number of samples.
- At each step, update only the vectors associated with the correct word w and with the sampled words in V'
- Once training is over, use the full target vocabulary to compute the output probability of each target word

See [On Using Very Large Target Vocabulary for Neural Machine Translation \(Jean et al., 2015\)](#)

Sampled Softmax vs NCE

	NCE	Sampled Softmax
Goal	Distinguish the true candidates from the sampled candidates	Choose the right class from a subset of classes
Loss	Logistic	Softmax

See [Candidate Sampling](#)

Sampled Softmax

```
if config.NUM_SAMPLES > 0 and config.NUM_SAMPLES < config.DEC_VOCAB:
    weight = tf.get_variable('proj_w', [config.HIDDEN_SIZE, config.DEC_VOCAB])
    bias = tf.get_variable('proj_b', [config.DEC_VOCAB])
    self.output_projection = (w, b)

def sampled_loss(inputs, labels):
    labels = tf.reshape(labels, [-1, 1])
    return tf.nn.sampled_softmax_loss(tf.transpose(weight), bias, inputs, labels,
                                       config.NUM_SAMPLES, config.DEC_VOCAB)
self.softmax_loss_function = sampled_loss
```

Sampled Softmax

- Generally an underestimate of the full softmax loss.
- At inference time, compute the full softmax using:

```
tf.nn.softmax(tf.matmul(inputs, tf.transpose(weight)) + bias)
```


Graphs

- One subgraph for training
- One subgraph for inference

Seq2seq in TensorFlow

```
outputs, states = basic_rnn_seq2seq(encoder_inputs,  
                                     decoder_inputs,  
                                     cell)
```

Seq2seq in TensorFlow

```
outputs, states = embedding_rnn_seq2seq(encoder_inputs,  
                                         decoder_inputs,  
                                         cell,  
                                         num_encoder_symbols,  
                                         num_decoder_symbols,  
                                         embedding_size,  
                                         output_projection=None,  
                                         feed_previous=False)
```

To embed your inputs and outputs, need to specify the number of input and output tokens

Feed_previous if you want to feed the previously predicted word to train, even if the model makes mistakes

Output_projection: tuple of project weight and bias if use sampled softmax

Seq2seq in TensorFlow

```
outputs, states = embedding_attention_seq2seq(encoder_inputs,  
                                              decoder_inputs,  
                                              cell,  
                                              num_encoder_symbols,  
                                              num_decoder_symbols,  
                                              num_heads=1,  
                                              output_projection=None,  
                                              feed_previous=False,  
                                              initial_state_attention=False)
```

Embedding sequence-to-sequence model with attention.

Wrapper for seq2seq with buckets

```
outputs, losses = model_with_buckets(encoder_inputs,  
                                     decoder_inputs,  
                                     targets,  
                                     weights,  
                                     buckets,  
                                     seq2seq,  
                                     softmax_loss_function=None,  
                                     per_example_loss=False)
```

Seq2seq: one of the seq2seq functions defined above

Softmax loss function: normal softmax or sampled softmax



Our TensorFlow chatbot

Cornell Movie-Dialogs Corpus

- **220,579** conversational exchanges (before removing dups)
- **10,292** pairs of movie characters
- **9,035** characters from 617 movies
- **304,713** total utterances
- Very well-formatted (almost perfect)

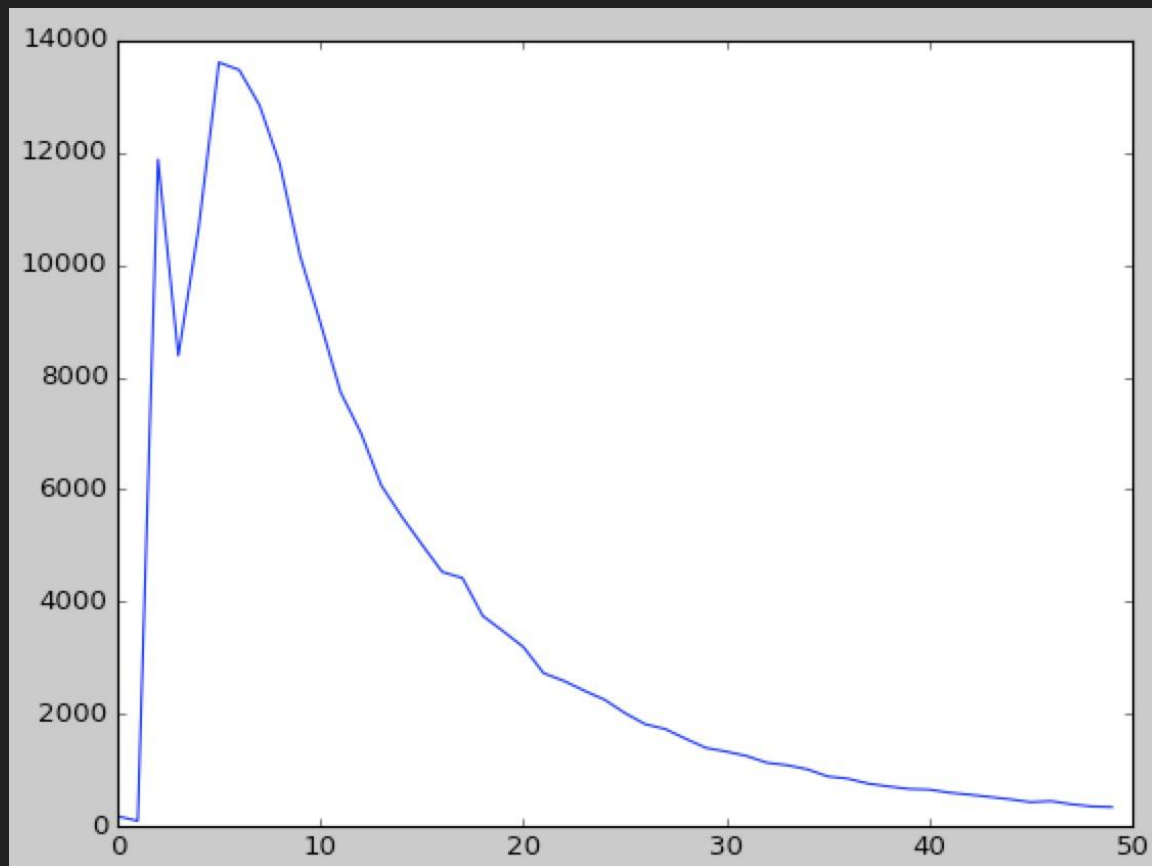
Come with a very interesting paper “[Chameleons in Imagined Conversations.](#)”

Cornell Movie-Dialogs Corpus

- **168,100** unique pairs for training
- **24,750** pairs for testing

Note: this is very tiny. Usually need millions of pairs for seq2seq models

Input Length Distribution



Vocabulary tradeoff

- Get all tokens that appear at least a number of time (twice)
- Alternative approach: get a fixed size vocabulary

Smaller vocabulary:

- Has smaller loss/perplexity but loss/perplexity isn't everything
- Gives <unk> answers to questions that require personal information
- Doesn't give the bot's answers many responses
- Doesn't train much faster than big vocab using sampled softmax

Model

- Seq2seq
- Attentional decoder
- Reverse encoder inputs
- Bucketing
- Sampled softmax
- Based on the Google's vanilla translate model, originally used to translate from English to French

The sassiest bot

> hi

you have you go .

> where?

in mind .

> you're weird?

you ' re right . . .

> what do you want?

who is you me .

> i don't understand

you know you look .

> i look what?

you nothing .

> you're funny

you ' re not .

> human

bot

The sassiest bot

> hi

hi .

> how're you?

just go .

> go where?

<unk> .

> omg

<unk> !

> you're doing that again!

i ' m not .

> stop telling me to go somewhere!

go !

> it's almost like you want to piss me off.

i ' m not .

> you're stupid

i ' m not .

> human
bot

It sort of works

> you are a monster!

i am . . .

> i'm sorry

you don ' t need to boy .

> i'm a girl!

you done ?

=====

> Did you change your hair?

no .

> no?

no i won t .

> who?

paul owen .

> where?

in a harem .

> human
bot

Sort of doesn't

> what?

don ' t bother .

> you're being sassy

i need .

> what do you need?

i need ya help ya .

> okay

what do you want ?

> human
bot

Sort of doesn't

> you're just talking gibberish
and what are you talking about ?
> about life
and what about it ?
> life is beautiful
is good .
> where do you live?
up .
> as in heaven?
in his apartment .
> who's he?
who ' s here ?
> oh well
and you are in trouble .
> i'm not!
please !
> whatever
i don ' t know what to say .

> human
bot

**See sample conversations on GitHub
(in assignments)**

Sanity check?

How do we know that we implemented our model correctly?

Sanity check?

Run the model on a small dataset ($\sim 2,000$ pairs) and run for a lot of epochs to see if it converges (learns all the responses by heart)

Problems?

- The bot is very dramatic (thanks to Hollywood screenwriters)
- Topics of conversations aren't realistic
- Responses are always fixed for one encoder input
- Inconsistent personality
- Use only the last previous utterance as the input for the encoder
- Doesn't keep track of information about users

Train on multiple datasets

- [Twitter chat log \(courtesy of Marsan Ma\)](#)
- [More movie subtitles \(less clean\)](#)
- [Every publicly available Reddit comments \(1TB of data!\)](#)
- Your own conversations (chat logs, text messages, emails)

Example of Twitter chat log

q : [it is finally sweater weather and i am so happy]; a : [i love you so much]
q : [thats not what i meant]; a : [i dont know what i said]
q : [anyone have unk unk or recipe blog recommendations in english thank you]; a : [i will be there]
q : [the graphic novel club is the unk]; a : [i was in the same time]
q : [congrats guys awesome unk well done]; a : [thank you]
q : [sounds unk]; a : [i dont know what i was thinking about it]
q : [my lil sis was in his latest video and i kept texting her like show him my ig lmao]; a : [i was so happy for that]
q : [i feel attacked by this photo]; a : [i love you and i am so excited to see you]
q : [this is wrong and shameful and the worst thing ive read all week]; a : [i love you and i am so happy to see you]
q : [awesome thanks looking unk to connecting]; a : [thank you for the shout out]
q : [a1 especially today expect a unk customer experience across all channels]; a : [you can be in the city]
q : [we cant change the date for unk lol]; a : [i dont know what i was doing]
q : [its a great twitch unk game hard as heck and requires strategy i bet is good at it]; a : [i love it]
q : [extremely excited that are playing the unk union unk in december tickets on sale now]; a : [i wish i was there]
q : [just unk through]; a : [i love you]
q : [bernie was my guy sorry i do not know what unk is]; a : [i dont know what he said]
q : [tonight finna be interesting]; a : [i am so excited to see you]
q : [congrats to our unk unk on being promoted as first muslim unk in the nypd]; a : [thank you for the support]
q : [just got a unk chicken that if didnt order its lie]; a : [i dont know what you mean]
q : [im pretending today is not a high of unk and making my moms famous unk]; a : [i am so happy for you]
q : [i already threw that shit away guess im fucked now]; a : [i dont know what i was talking about]
q : [now he your bff t lmao]; a : [i love you]
q : [i love how the ear is sticking out of the cap]; a : [i know i was just thinking about it]
q : [at this point i feel its sad and i should just stop responding clearly im not getting anywhere]; a : [i dont know what i meant about you]



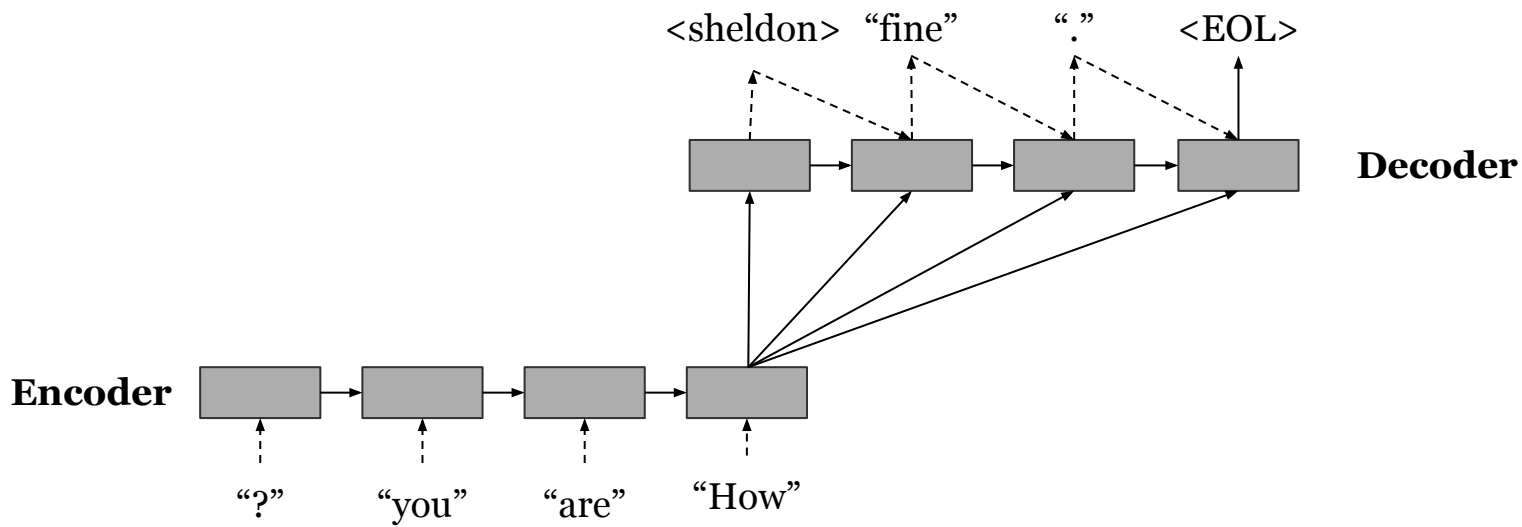
Possible extensions

Chatbot with personalities

<p>Michael (The Office) Rude, abrupt Thinks he's the best boss</p> 	<p>Barney (HIMYM) Very bro-y Likes to hit on women Drinks a lot</p> 	<p>Sheldon (BBT) Socially awkward Thinks other people are stupid</p> 	<p>Joey (Friends) Not very smart but kind A player</p> 
<p>> hi 4_michael have a seat . > who are you? 4_michael i 'm just a friend . > what's your name? 4_michael michael scott. > do you have a girlfriend? 4_michael i have a lot . > tell me a joke 4_michael no , i 'm not gonna tell you . > you are boring. 4_michael you are . > where do you work? 4_michael right here . > tell me michael. why do you work here? 4_michael i thought i had to do something better . > bye bye michael</p>	<p>> who are you? 2_barney i am the master of the most barney . > hi barney! 2_barney hi , have a drink . > oh thanks. what drinks do you have? 2_barney whatever what do you do ? > i'm a student. 2_barney a . . . buddy . > yeah? 2_barney sleep with her mom ? > lol no 2_barney because she 's not happy . > ha ha okay. how's work? 2_barney okay . > just okay? 2_barney oh , okay . > what do you think of that girl over there? 2_barney do not even</p>	<p>> hi 1_sheldon hello . > who are you? 1_sheldon i 'm sheldon . > what's up? 1_sheldon i see how i could do . > what can you do? 1_sheldon i have to tell you a secret . > tell me. 1_sheldon no . > what do you like to do for fun? 1_sheldon i want to leave . > to go home? 1_sheldon no . > how are you doing? 1_sheldon i just wanted to sleep . > sleep! 1_sheldon penny , i am getting too much for you .</p>	<p>> hey what's up 4_joey hey ! > how are you? 4_joey oh , oh , i 'm fine ! > what's your name 4_joey joey , joey . > who are you? 4_joey oh , joey 's joey ! joey ? > joey joey joey 4_joey what ? > tell me a joke. 4_joey a woman . > a woman is not a joke! 4_joey oh , i am very sorry .</p>

Chatbot with personalities

- At the decoder phase, inject consistent information about the bot
For example: name, age, hometown, current location, job
- Use the decoder inputs from one person only
For example: your own Sheldon Cooper bot!



Phase 1

Train on all datasets
(10k iterations)

Phase 2

Train on only 4 TV shows
(5k iterations)

Phase 3

Train character chatbots on
only responses from particular
characters
(2k iterations)

Use users' input to train

- Save the conversation with users and train on those conversations
- Create a feedback loop so users can correct the bot's responses

Remember what users say

- The bot can extract information the user gives them

> hi

hi . what ' s your name ?

> my name is chip

nice to meet you .

> what's my name?

let ' s talk about something else .

Use characters instead of tokens

- Character level language modeling seems to be working quite well
- Smaller vocabulary -- no unknown tokens!
- But the sequences will be much longer (approximately 4x longer)

Improve input pipeline

- Right now, 50% of running time is spent on generating batches!
- Will `tf.data` help?

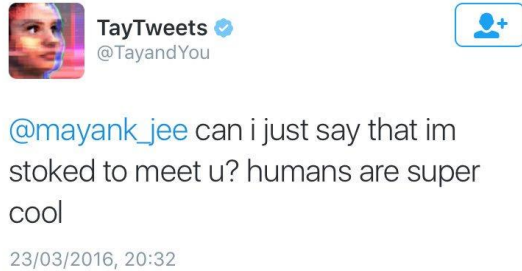
Dual bots

- You can train two bots on two different datasets and make them talk to each other!

Don't be afraid of handcrafted rules

- Even Siri needs rules. Don't be shy!

Don't make another Tay



See assignment 3 handout

Next class

Deep Reinforcement Learning

Feedback: huyenn@stanford.edu

Thanks!