香港城市大學（東莞）
City University of Hong Kong
(Dongguan)

Celebrating the Establishment in Year 2024

Lecture 6: NLP tasks (2) Generation tasks

CS6493 Natural Language Processing

Instructor: Linqi Song

Outline

- Natural language generation tasks
- Machine translation
  - What is machine translation?
  - Neural machine translation
  - Evaluation
- Dialogue system
  - What is a dialogue system?
  - Task-oriented dialogue systems
  - Chitchat dialogue systems
  - Performance evaluation

Natural language generation tasks

- NLG focuses on generating human-like text that conveys information or communicates effectively. It involves tasks such as machine translation and dialogue generation.
- Generation tasks
  - Machine translation
  - Paraphrasing (rewriting)
  - Report generation & long text generation
  - Summarization
  - Dialog generation
  - …

Machine Translation

What is machine translation (MT)?

- Translate a sentence x from one language (the source language) to a sentence y in another language (the target language).

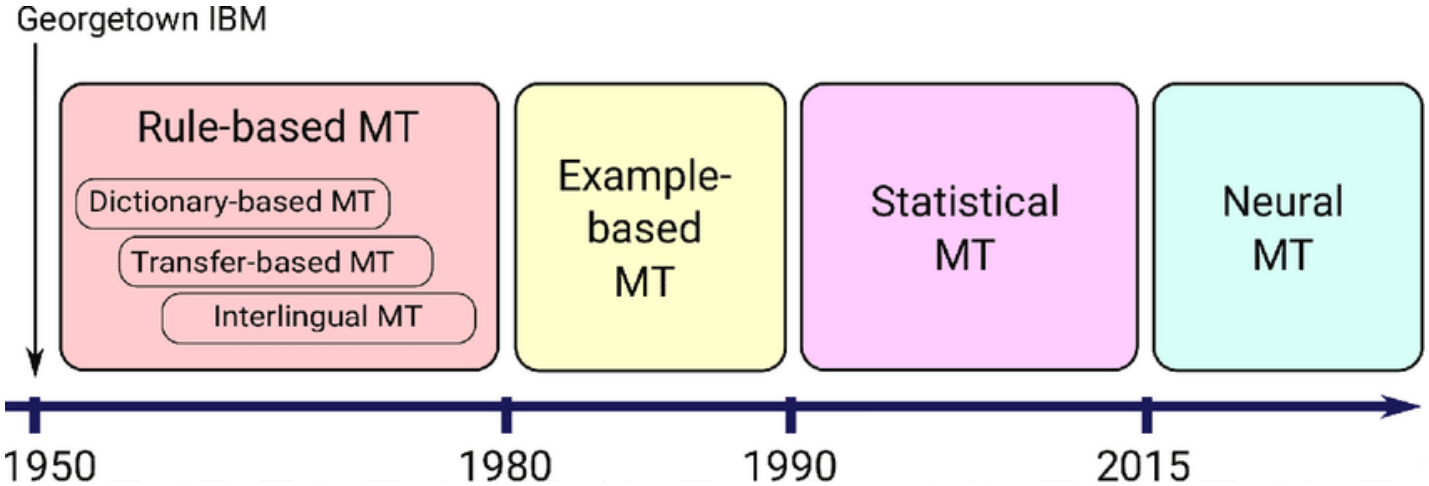
x: *L'homme est né libre, et partout il est dans les fers*

y: *Man is born free, but everywhere he is in chains*

- Rousseau

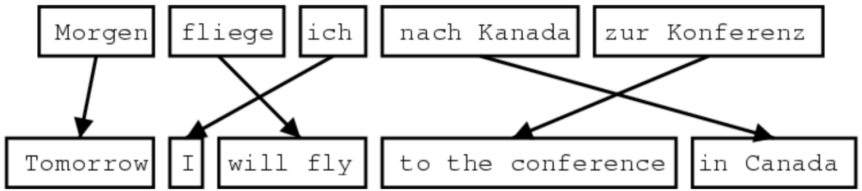
Timeline of machine translation

- 1950s: rule-based MT
- 1980s-1990s: example-based MT
- 1990s - 2010s: statistical machine translation (SMT)
- Current trend: neural machine translation (NMT)



Statistical MT model

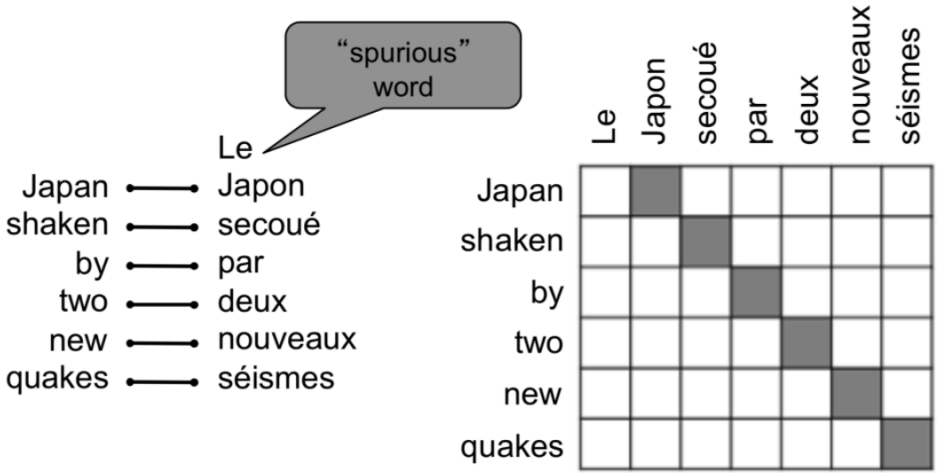
- How to Learn translation model $P(x|y)$?
  Need large amount of parallel data
  (e.g. pairs of human-translated German / English sentences).



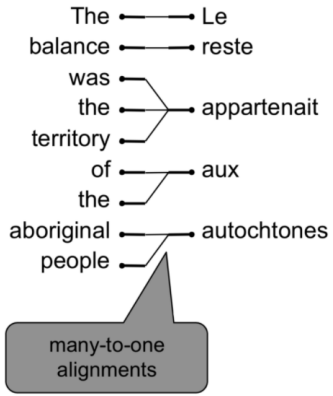
- How to learn $P(x|y)$ from the parallel corpus?
  - Introduce a latent variable a : $P(x, a|y)$
  - a is the alignment: word-level correspondence between source sentence x and target sentence y

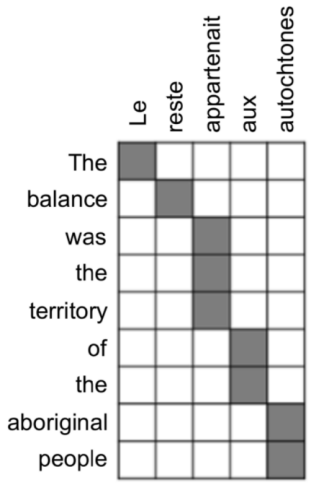
Alignemnt details

- Alignment is the correspondence between particular words in the translated sentence pair.
- Typological differences between languages lead to complicated alignments.
- Some words have **no** counterpart.

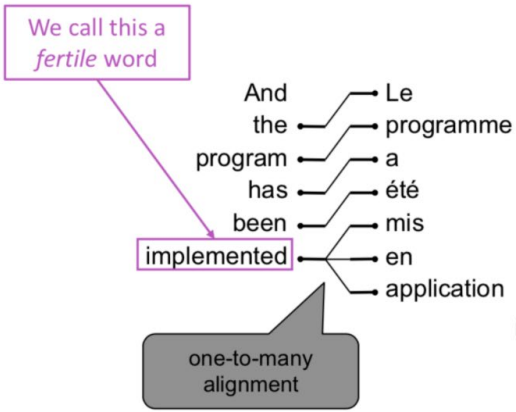


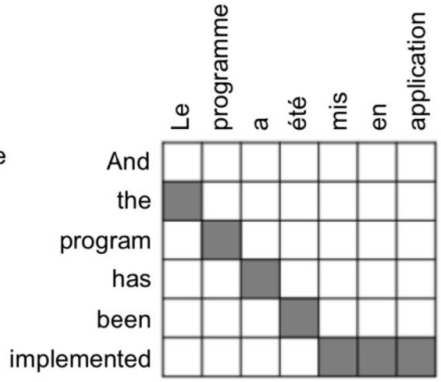
Many-to-one Alignment



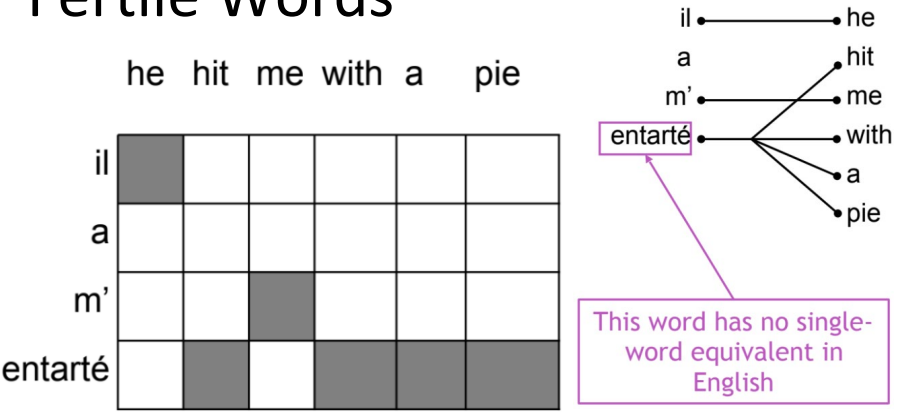


One-to-many Alignment

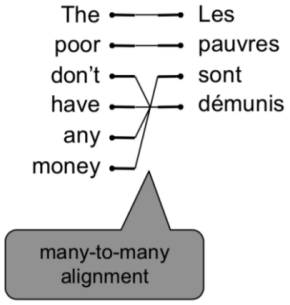


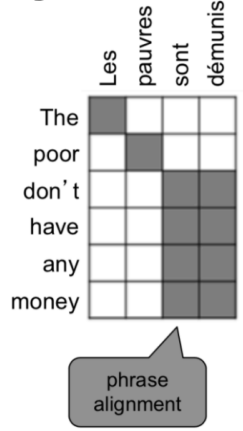


Fertile Words



Many-to-many Alignment (phrase level)





NMT: the biggest success story of NLP deep learning

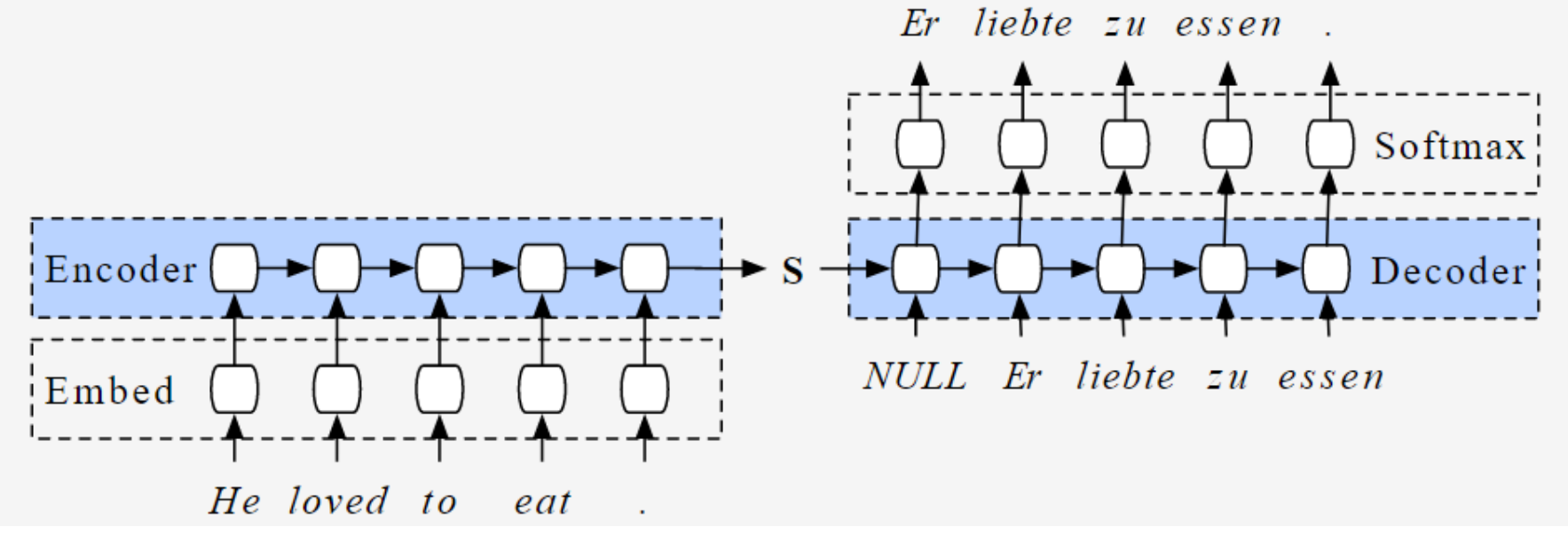
Neural Machine Translation went from a fringe research activity in 2014 to the leading standard method in 2016.

- **2014**: First seq2seq paper published
- **2016**: Google Translate switches from SMT to NMT

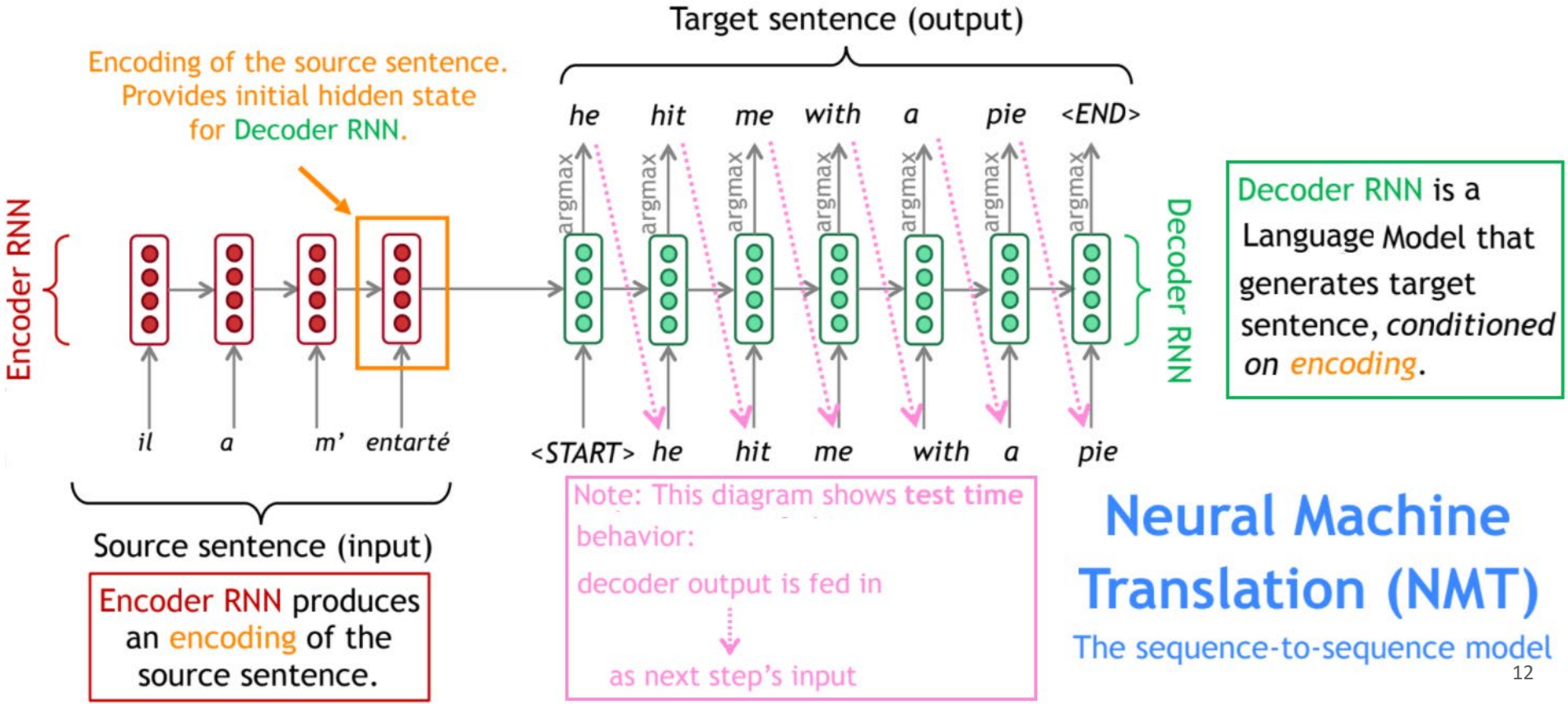
SMT systems, built by hundreds of engineers *over many years*, outperformed by NMT systems trained by a handful of engineers in *a few months*.

Seq2seq model

Sequence-to-sequence (seq2seq) is an encoder-decoder neural network architecture that is to convert sequences from one domain (e.g. sentences in English) to another, and it involves two RNNs: encoder RNNs and decoder RNNs.



NMT with seq2seq model



Neural Machine Translation (NMT)

The sequence-to-sequence model

Seq2seq as a conditional language model (1)

The seq2seq model is an example of a Conditional Language Model.

- 'Language model': because the decoder is predicting the next word of the target sentence y
- 'Conditional': because its predictions are also conditioned on the source sentence x

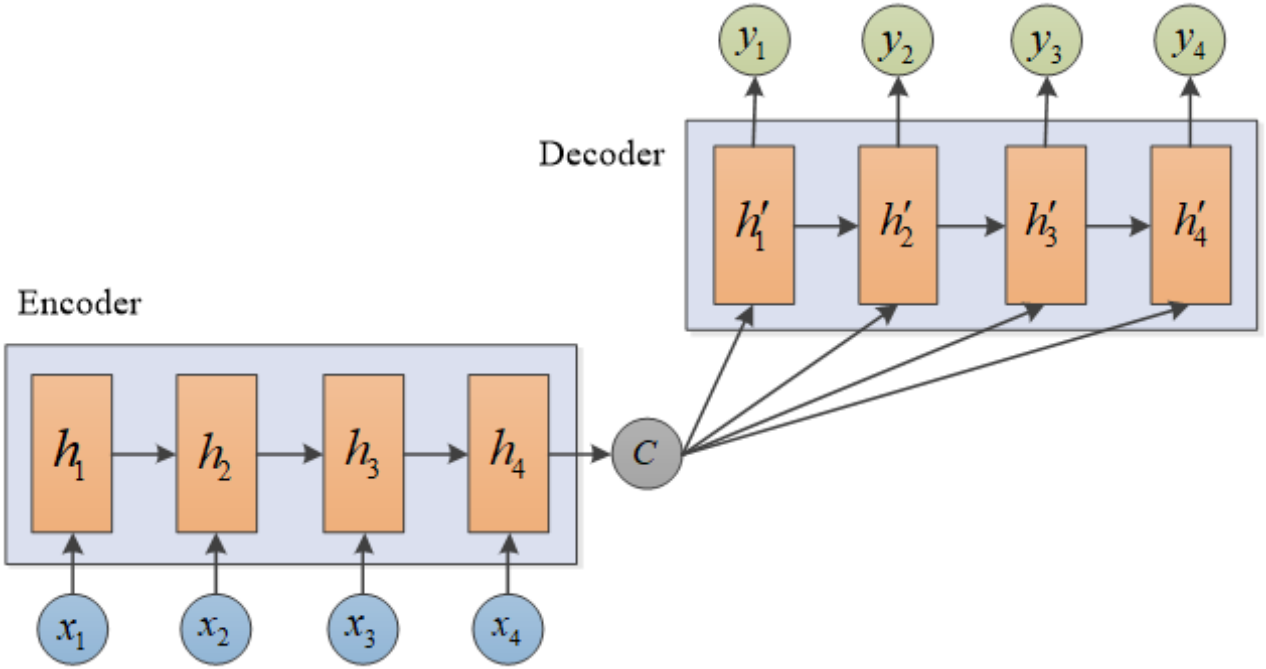
NMT directly calculates $P(y|x)$

$$P(y|x) = P(y_1|x)\,P(y_2|y_1,x)\,P(y_3|y_1,y_2,x)\ldots\underbrace{P(y_T|y_1,\ldots,y_{T-1},x)}$$

Probability of next target word, given target words so far and source sentence x

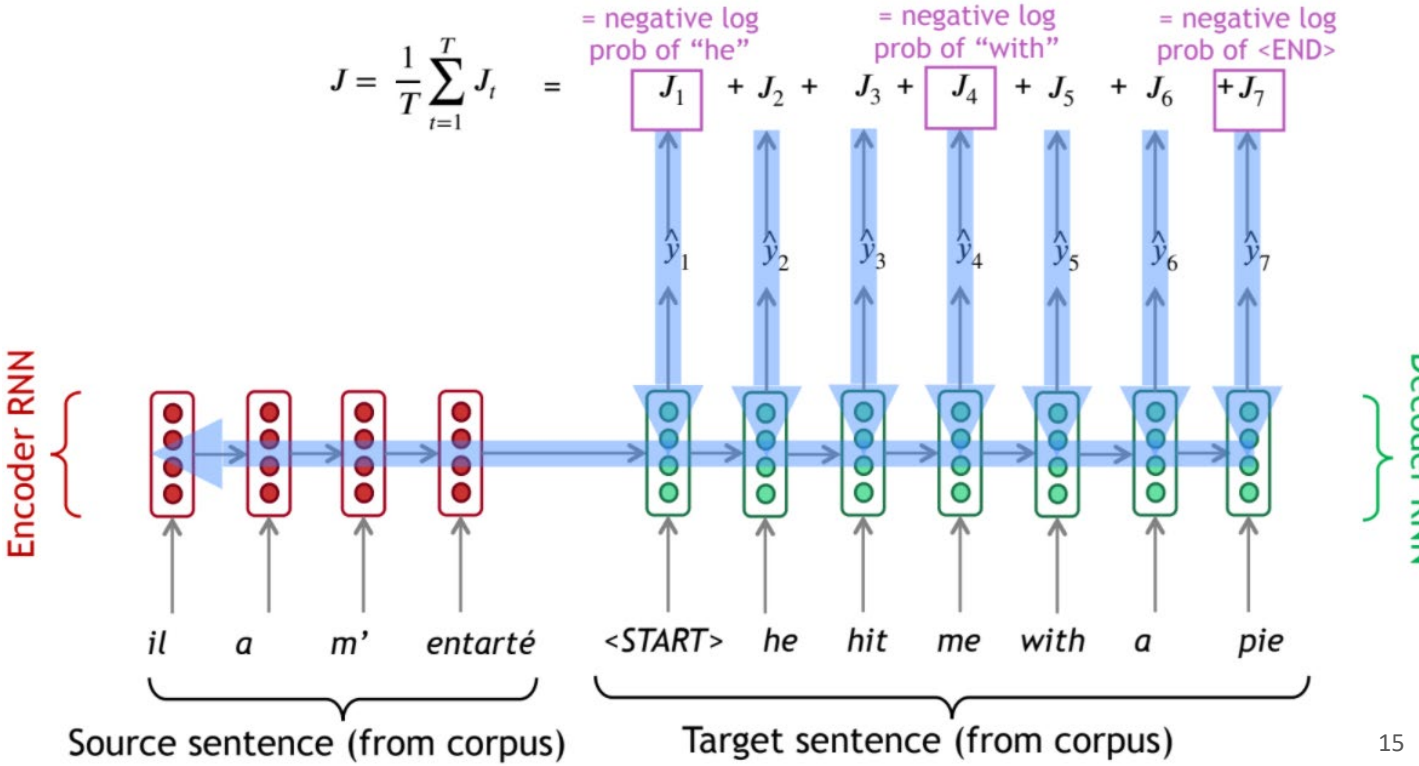
Seq2seq as a conditional language model (2)

Conditional on the hidden state C, which is the output of the encoder RNNs.



Train a seq2seq NMT system

Seq2seq is optimized as a single system. Backpropagation operates "end-to-end".



Greedy decoding

- Decode the target sentence by taking argmax on each step of the decoder
- This method is equivalent to taking the most probable word on each step.
- Problem: no way to undo decisions.

Input: *il a m'entarté* *(he hit me with a pie)*

→ *he* ____

→ *he hit* ____

→ *he hit a* ____ (whoops! no going back now…)

Exhaustive search decoding

- Instead of maxmizing the probability in each step, we hope to find a translation y with length T such that

$$P(y|x) = P(y_1|x)\,P(y_2|y_1,x)\,P(y_3|y_1,y_2,x)\ldots,P(y_T|y_1,\ldots,y_{T-1},x)$$
$$= \prod_{t=1}^{T} P(y_t|y_1,\ldots,y_{t-1},x)$$

  is maximized.
- **Problem**: computing all possible sequences y has a complexity of $O(V^T)$.
  On each step t of the decoding, we are tracking V^t possible partial translations, where V is vocabulary size.

Beam search decoding

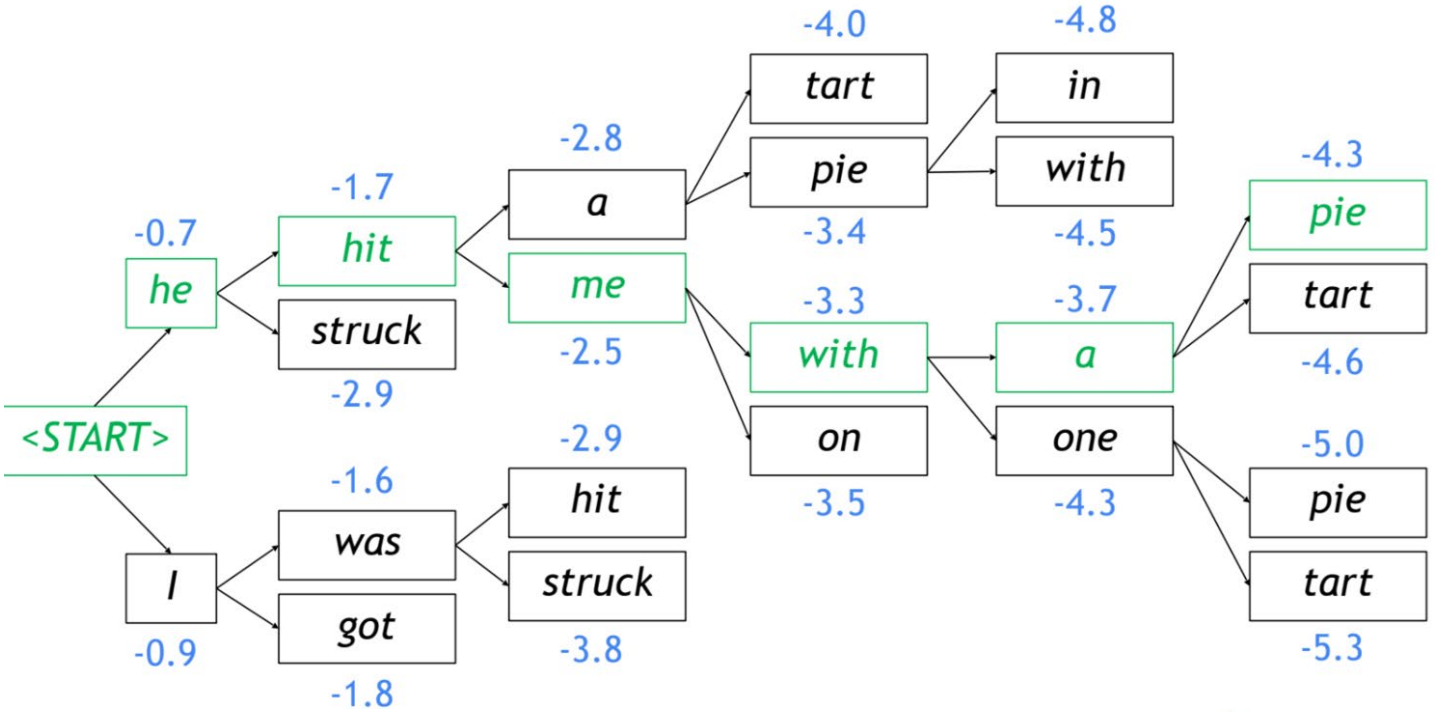
- On each step of decoding, keep track of the k most probable partial translations (k hypotheses)
- k is the beam size (in practice around 5 to 10)
- A hypothesis $(y_1, y_2, \dots, y_t)$ is scored by its log probability:

$$\text{score}(y_1, \dots, y_t) = \log P_{\text{LM}}(y_1, \dots, y_t | x) = \sum_{i=1}^{t} \log P_{\text{LM}}(y_i | y_1, \dots, y_{i-1}, x)$$

  Scores are all negative, and higher scores are better.
- Longer hypotheses have lower scores, so normalized by length:

$$\frac{1}{t} \sum_{i=1}^{t} \log P_{\text{LM}}(y_i | y_1, \dots, y_{i-1}, x)$$

A beam search example with $k = 2$



Blue numbers $= \text{score}(y_1, \ldots, y_t) = \sum_{i=1}^{t} \log P_{\text{LM}}(y_i | y_1, \ldots, y_{i-1}, x)$

Stopping criteria for beam search

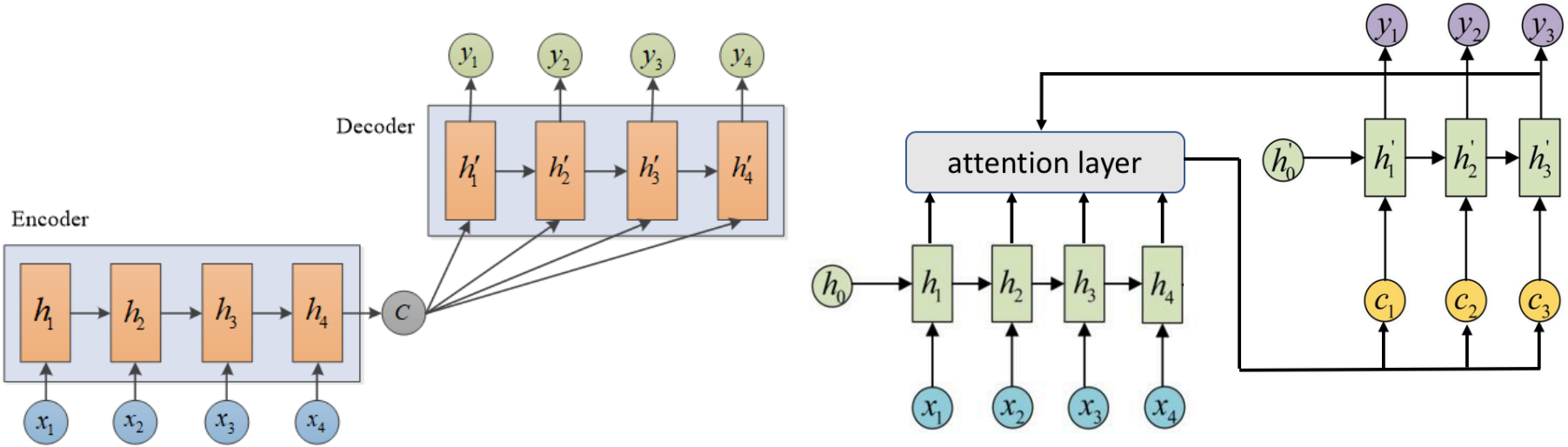
In beam search decoding, different hypotheses may produce <END> tokens on different timesteps

- When a hypothesis produces <END>, that hypothesis is complete.
- Place it aside and continue exploring other hypotheses via beam search.
- Usually we continue beam search until:
  - We reach timestep T (where T is some pre-defined cutoff), or
  - We have at least n completed hypotheses (where n is a pre-defined cutoff)

Seq2seq with vs. without attention

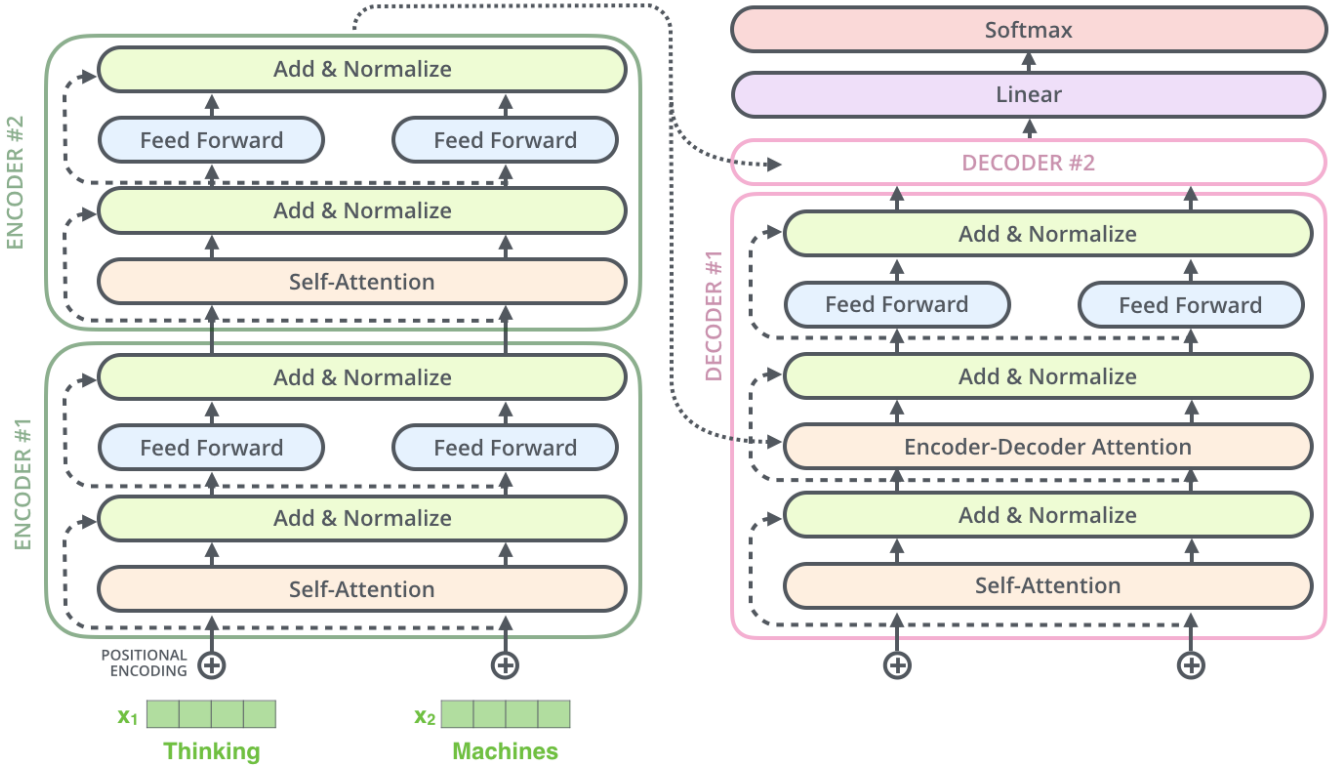
Without attention: conditioned on the same context (hidden state) C

With attention: conditioned on varying context (hidden state) $c_1, c_2, c_3, \ldots$



Transformer-based MT

Encoder-decoder structure, multi-head self-attention, decoding similar to seq2seq



MT model evaluation – BLEU (1)

BLEU (Bilingual Evaluation Understudy)

- BLEU compares the machine-written translation (candidate sentence) to one or several human-written translation(s) (reference sentences), and computes a similarity score based on:
  - n-gram precision p_n (usually for 1, 2, 3 and 4-grams)
  - A brevity penalty for too-short system translations

$$\mathrm{BP} = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases} .$$

(c: candidate sentence length, r: reference sentence length)

Then,

$$\mathrm{BLEU} = \mathrm{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) .$$

(*n*-gram precision p_n, with weights w_n)

MT model evaluation – BLEU (2)

Precision calculation

Candidate sentence:
It is a guide to action which ensures that the military always obeys the commands of the party (18 words)

Reference sentences (gold standard)
1: It is a guide to action that ensures that the military will forever heed Party commands (16 words)
2: It is the guiding principle which guarantees the military forces always being under the command of the Party (18 words)
3: It is the practical guide for the army always to heed the directions of the party (16 words)

MT model evaluation – BLEU (3)

1-gram precision calculation p1 = sum(Min)/sum(Candidate) = 0.95

1-gram	Candidate	R1	R2	R3	Max_ref = Max (R1, R2, R3)	Min = Min (Candidate, Max_ref)
the	3	1	4	4	4	3
obeys	1	0	0	0	0	0
a	1	1	0	0	1	1
which	1	0	1	0	1	1
ensures	1	1	0	0	1	1
guide	1	1	0	1	1	1
always	1	0	1	1	1	1
is	1	1	1	1	1	1
of	1	0	1	1	1	1
to	1	1	0	1	1	1
commands	1	1	0	0	1	1
that	1	2	0	0	2	1
It	1	1	1	1	1	1
action	1	1	0	0	1	1
party	1	0	0	1	1	1
military	1	1	1	0	1	1

MT model evaluation – BLEU (4)

2-gram precision calculation p2 = sum(Min)/sum(Candidate) = 0.588235294

2-gram	Candidate	R1	R2	R3	Max_ref = Max (R1, R2, R3)	Min = Min (Candidate, Max_ref)
ensures that	1	1	0	0	1	1
guide to	1	1	0	0	1	1
which ensures	1	0	0	0	0	0
obeys the	1	0	0	0	0	0
commands of	1	0	0	0	0	0
that the	1	1	0	0	1	1
a guide	1	1	0	0	1	1
of the	1	0	1	1	1	1
always obeys	1	0	0	0	0	0
the commands	1	0	0	0	0	0
to action	1	1	0	0	1	1
the party	1	0	0	1	1	1
is a	1	1	0	0	1	1
action which	1	0	0	0	0	0
It is	1	1	1	1	1	1
military always	1	0	0	0	0	0
the military	1	1	1	0	1	1

MT model evaluation – BLEU (5)

3-gram precision calculation p3 = sum(Min)/sum(Candidate) = 0.4375

3-gram	Candidate	R1	R2	R3	Max_ref = Max (R1, R2, R3)	Min = Min (Candidate, Max_ref)
ensures that the	1	1	0	0	1	1
which ensures that	1	0	0	0	0	0
action which ensures	1	0	0	0	0	0
a guide to	1	1	0	0	1	1
military always obeys	1	0	0	0	0	0
the commands of	1	0	0	0	0	0
commands of the	1	0	0	0	0	0
to action which	1	0	0	0	0	0
the military always	1	0	0	0	0	0
obeys the commands	1	0	0	0	0	0
It is a	1	1	0	0	1	1
of the party	1	0	0	1	1	1
is a guide	1	1	0	0	1	1
that the military	1	1	0	0	1	1
always obeys the	1	0	0	0	0	0
guide to action	1	1	0	0	1	1

MT model evaluation – BLEU (6)

4-gram precision calculation p4 = sum(Min)/sum(Candidate) = 0.266666667

4-gram	Candidate	R1	R2	R3	Max_ref = Max (R1, R2, R3)	Min = Min (Candidate, Max_ref)
to action which ensures	1	0	0	0	0	0
action which ensures that	1	0	0	0	0	0
guide to action which	1	0	0	0	0	0
obeys the commands of	1	0	0	0	0	0
which ensures that the	1	0	0	0	0	0
commands of the party	1	0	0	0	0	0
ensures that the military	1	1	0	0	1	1
a guide to action	1	1	0	0	1	1
always obeys the commands	1	0	0	0	0	0
that the military always	1	0	0	0	0	0
the commands of the	1	0	0	0	0	0
the military always obeys	1	0	0	0	0	0
military always obeys the	1	0	0	0	0	0
is a guide to	1	1	0	0	1	1
It is a guide	1	1	0	0	1	1

MT model evaluation – BLEU (7)

1, 2, 3, 4-gram precisions with equal weights $\exp \sum_i 0.25 * \log p_i$=0.504566684

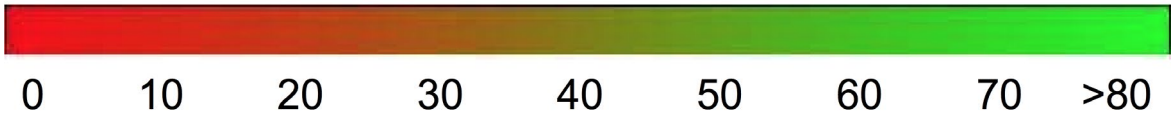
brevity penalty BP = 1 ($r = c = 18$), choosing the nearest reference sentence length among all references

BLUE = 0.504566684, pretty good translations!

MT model evaluation – BLEU (8)

- BLEU score interpretation

< 10	Almost useless
10 - 19	Hard to get the gist
20 - 29	The gist is clear, but has significant grammatical errors
30 - 40	Understandable to good translations
40 - 50	High quality translations
50 - 60	Very high quality, adequate, and fluent translations
> 60	Quality often better than human



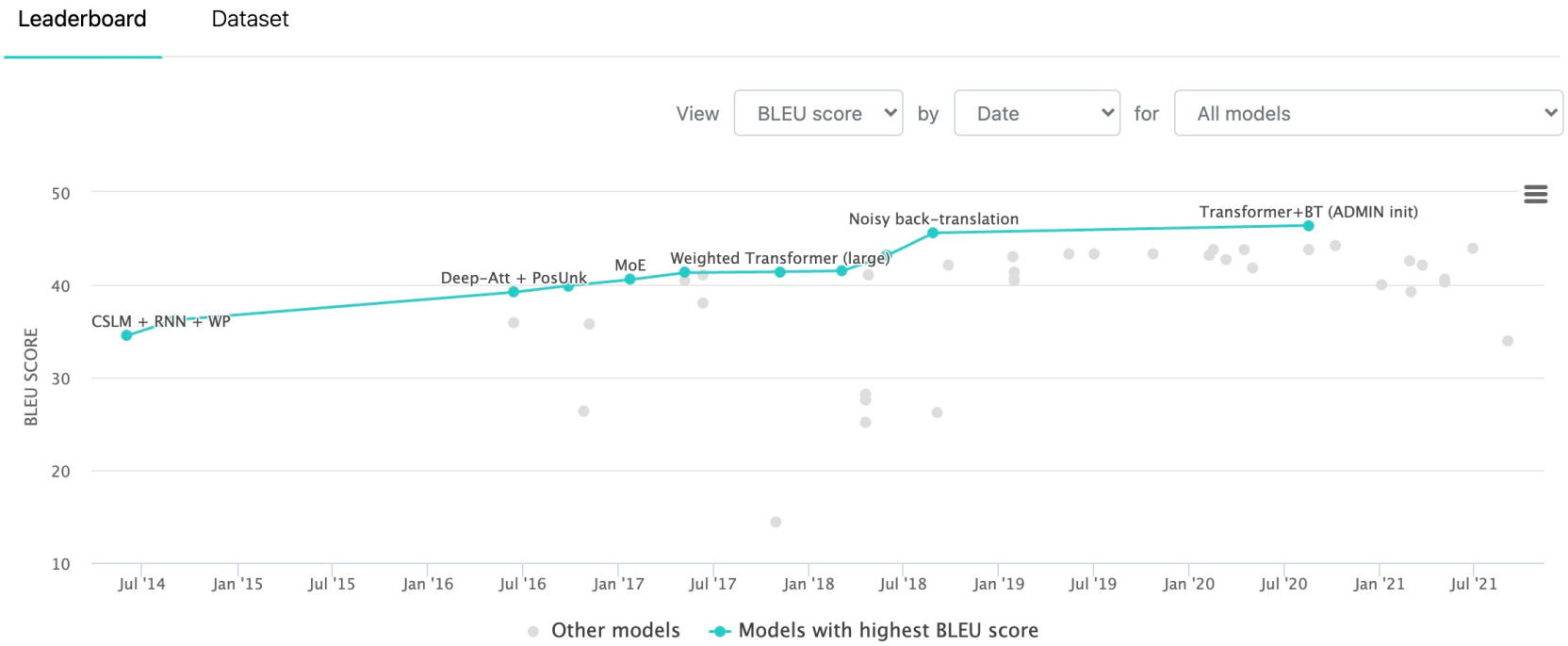
Comparing BLEU scores across different corpora and languages is strongly discouraged!

MT model evaluation – BLEU (9)

- BLEU is useful but imperfect
  - There are many valid ways to translate a sentence
  - So a good translation can get a poor BLEU score because it has low n-gram overlap with the human translation

NMT performance (1)

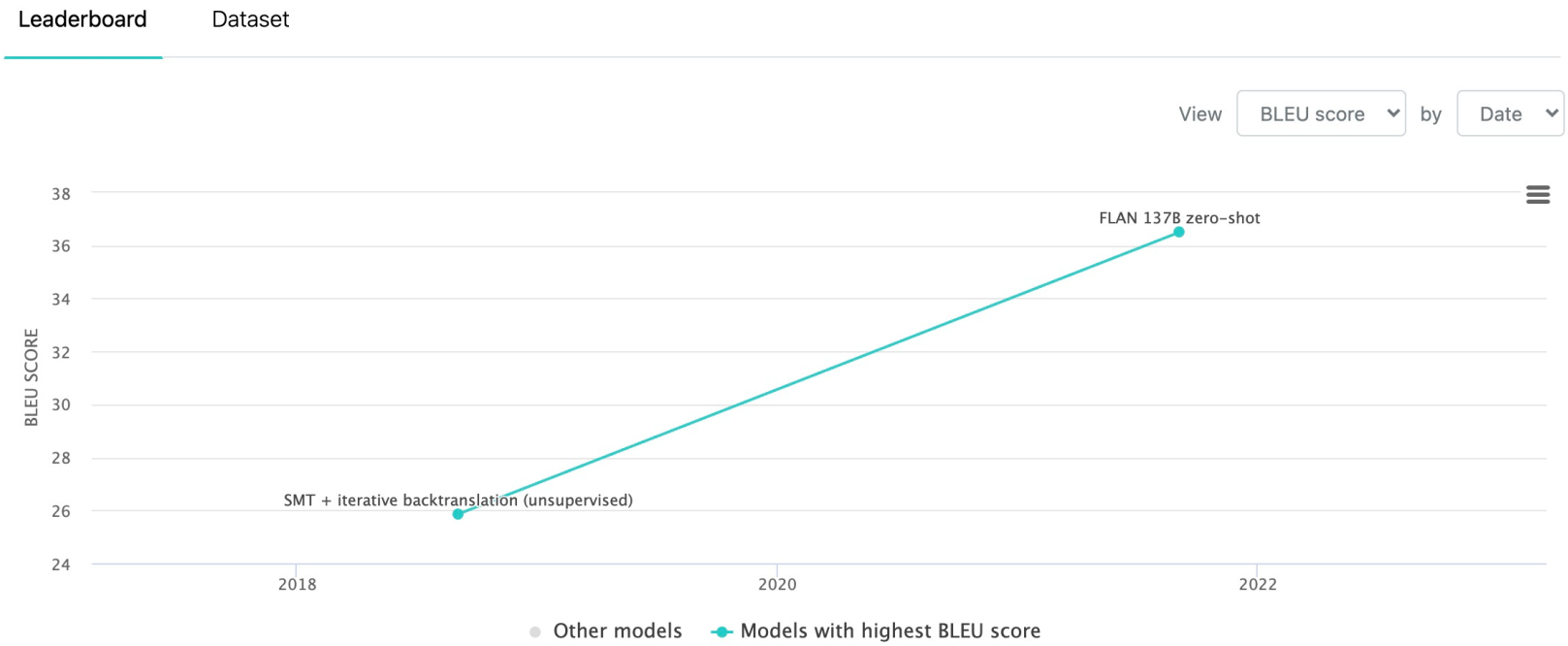
Machine Translation on WMT2014 English-French



https://paperswithcode.com/dataset/wmt-2014

NMT performance (2)

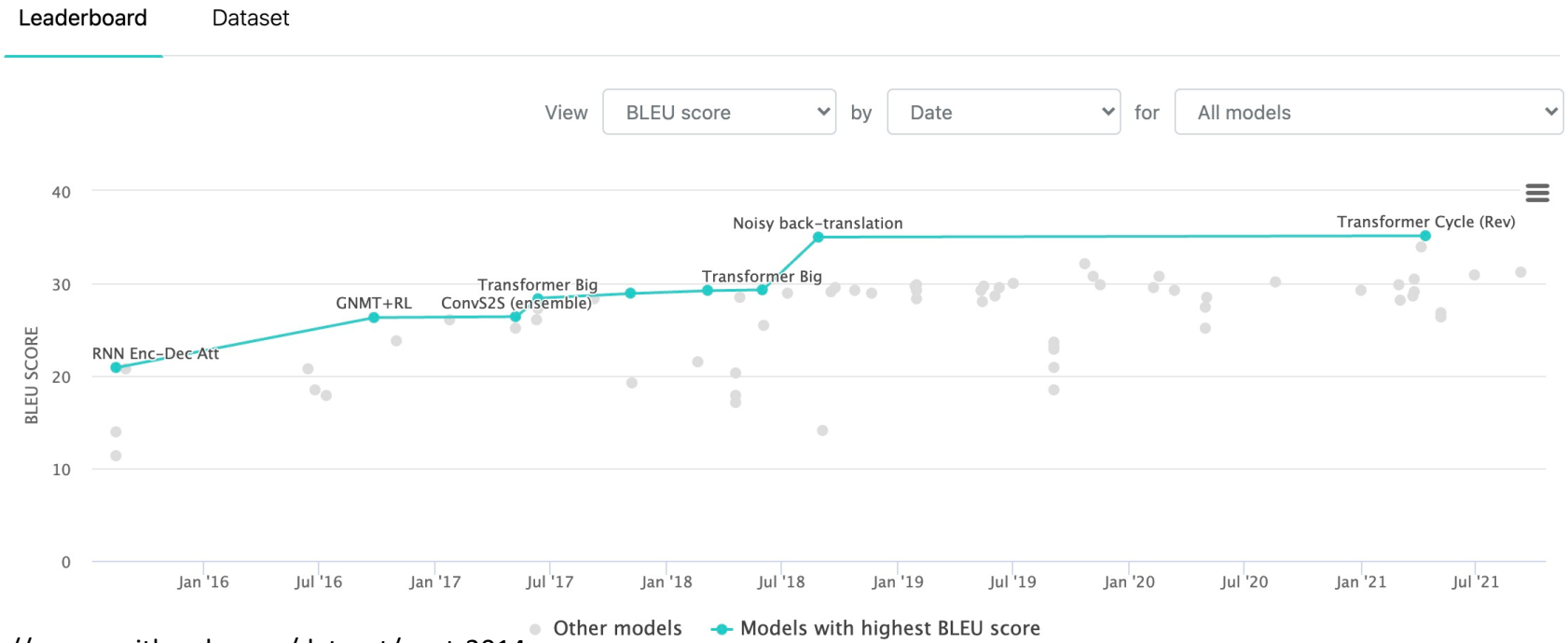
Machine Translation on WMT2014 French-English



https://paperswithcode.com/dataset/wmt-2014

NMT performance (3)

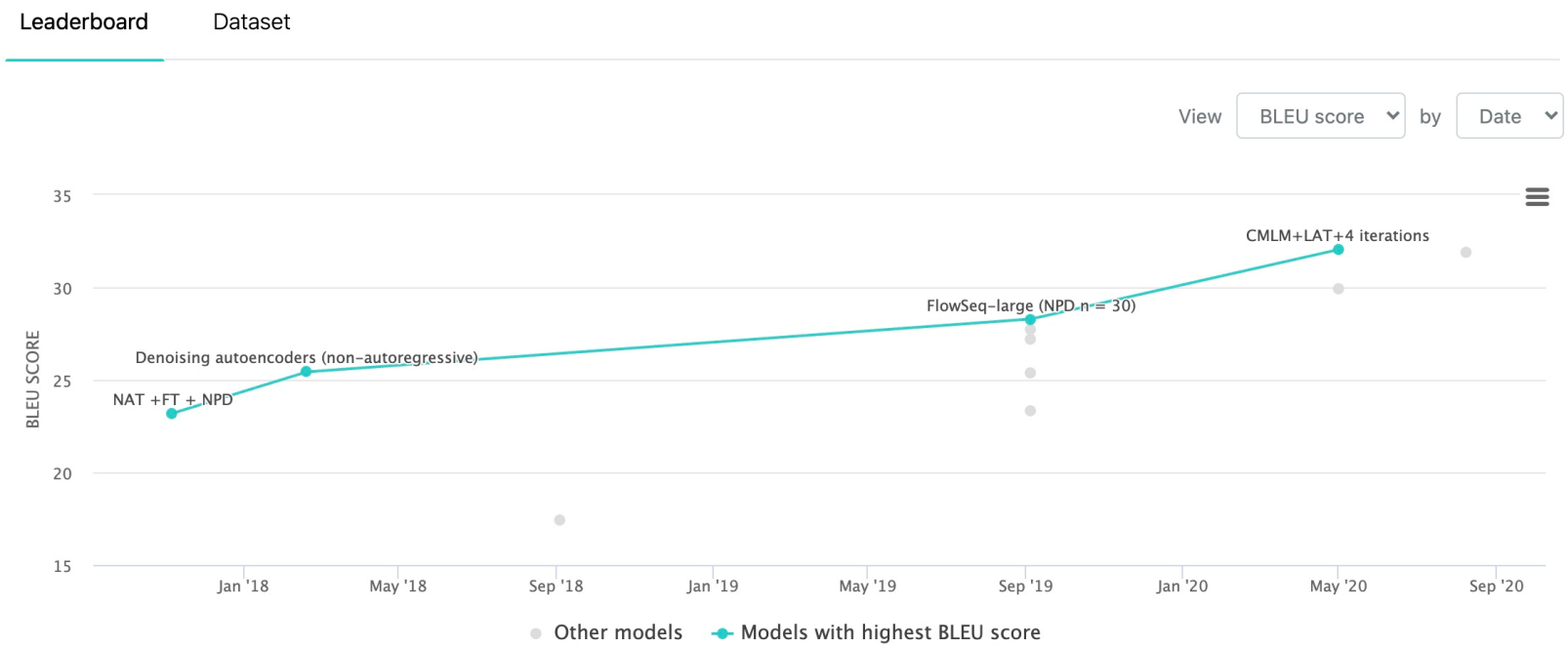
Machine Translation on WMT2014 English-German



https://paperswithcode.com/dataset/wmt-2014

NMT performance (4)

Machine Translation on WMT2014 German-English



https://paperswithcode.com/dataset/wmt-2014

Remaining challenges in MT

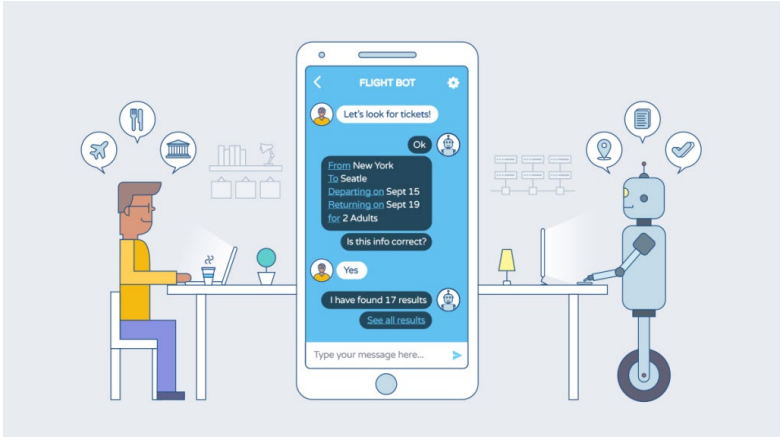
- Out-of-vocabulary words
- Maintaining context over longer text
- Low-resource language pairs
- Using common sense is still hard
- Idioms are difficult to translate

Dialog Systems

What is a dialogue system?

- A dialogue system is a computer system intended to converse with a human.
  - It employed one or more of text, speech, graphics, haptics, gestures, and other modes for communication on both the input and output channel.
  - We focus on natural language-based dialogue systems for different purposes (e.g., obtaining knowledge, booking tickets, casual chat, etc.)

Spoken dialogue systems are being incorporated into various devices (smart-phones, smart TVs, in-car navigating system, etc.).



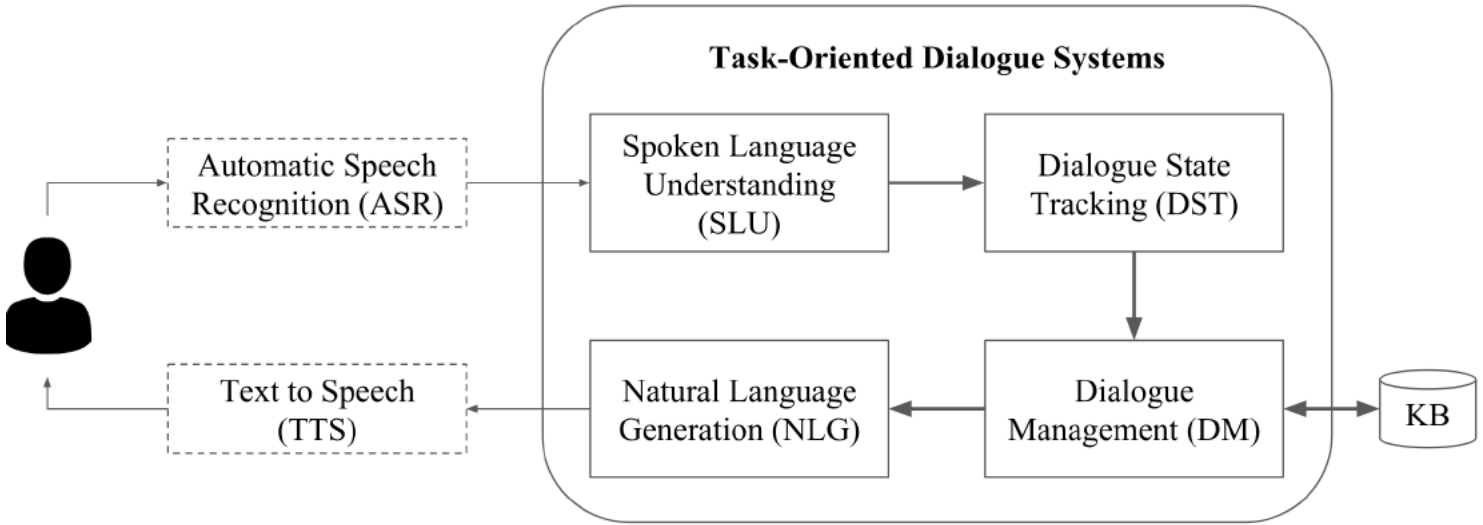
Text-based chatbot

Categories of dialog tasks

- Task-oriented
  - open- or close- domain
  - aim at recognize the task of the user and execute corresponding tasks to accomplish the goal
  - E.g., booking a restaurant, booking movie tickets, checking account balance, etc.
- Chitchat
  - open-domain
  - aim at respond to the user input in a conversational manner
  - E.g., making socially engaging conversations

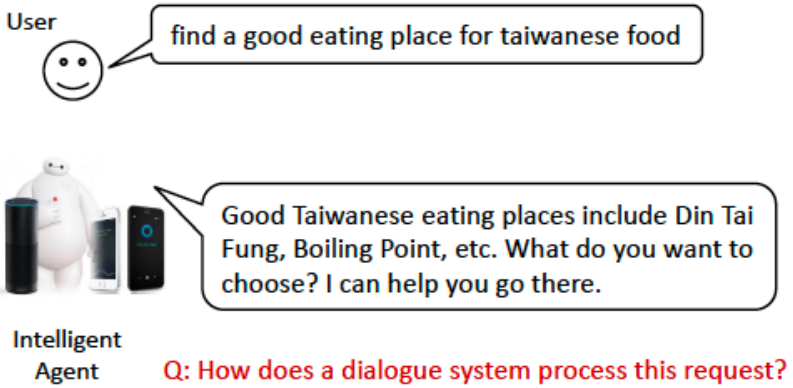
Task-oriented dialogue systems

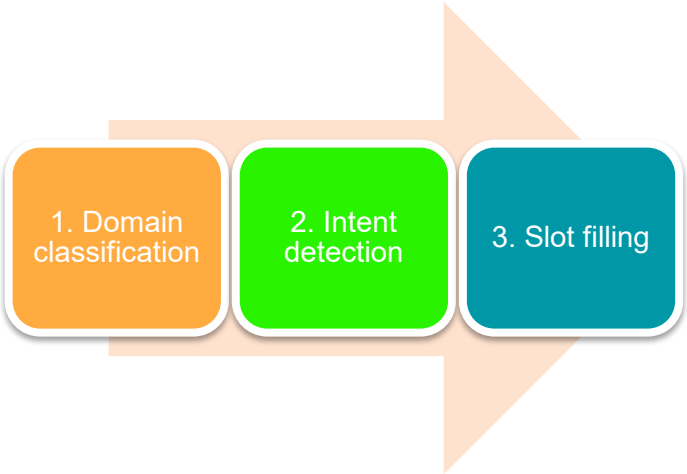
ASR (optional) -> SLU (NLU) -> DST -> DM (<-> knowledge base) -> NLG -> TTS (optional)



NLU in dialogue systems

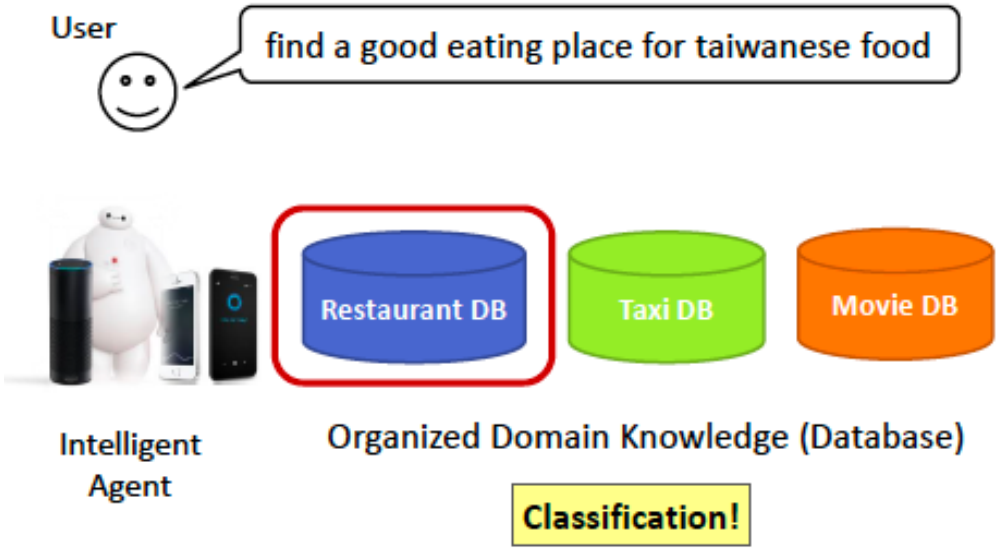
Pipelined tasks





Domain classification

Requires predefined domain ontology (ontologies usually describe a relationship between two concepts or entities).

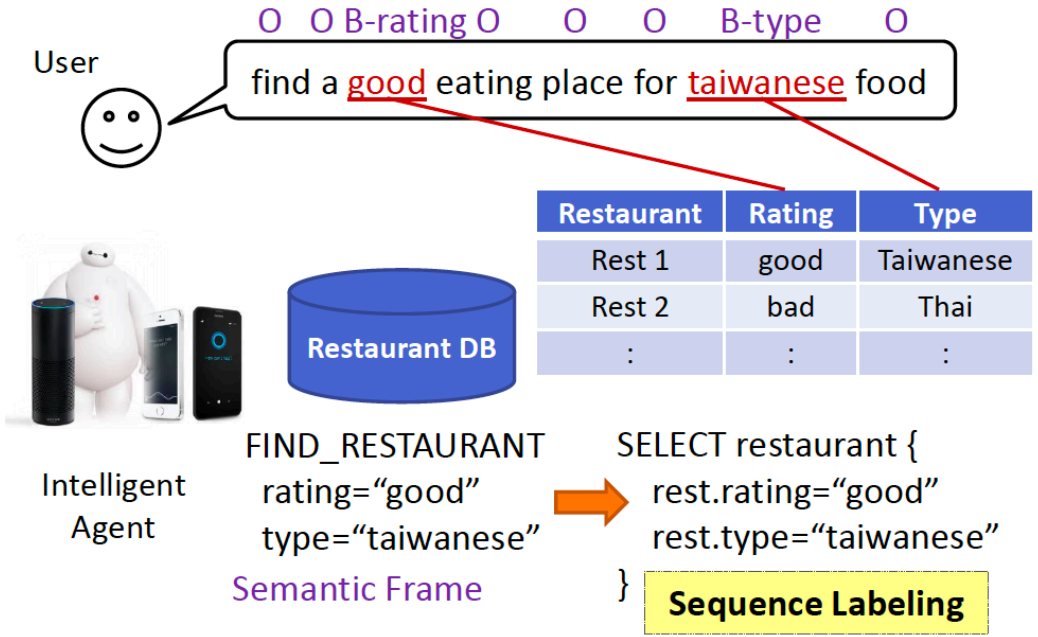


Intent detection

Requires predefined schema (the targeted information needs of domain experts by providing a user interface).

Slot filling

Requires predefined schema (the targeted information needs of domain experts by providing a user interface).



BIO (beginning, inside, outside) tagging:

I- prefix before a tag indicates that the tag is inside a chunk.

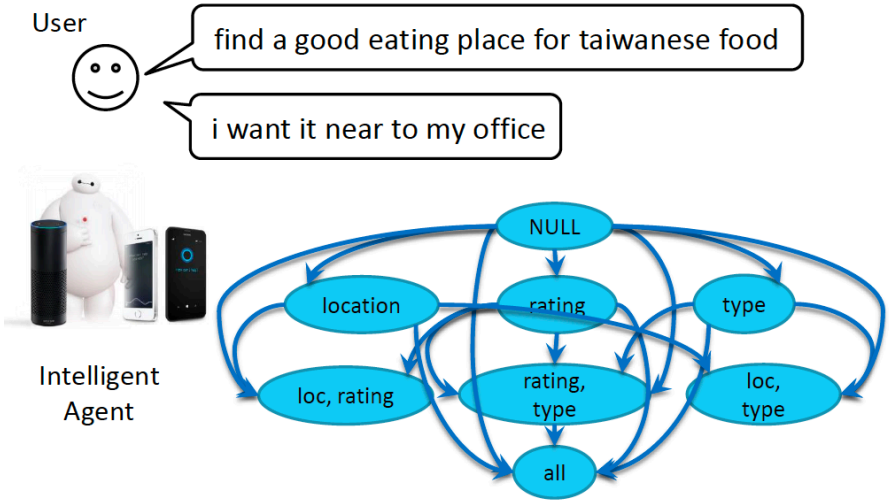
An O tag indicates that a token belongs to no chunk.

The B- prefix before a tag indicates that the tag is the beginning of a chunk.

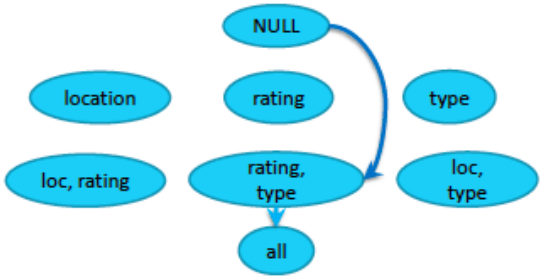
Dialogue state tracking (DST)

Requires handcrafted state machines (states, inputs, transition, outputs), like in a reinforcement learning scenario.

Observe the current dialogue state.



FIND_RESTAURANT
rating="good"
type="taiwanese"



Dialogue policy learning (DLP) or dialogue management (DM) for agent action

- DLP: determine what actions to choose.
- Enough information to generate an output utterance (location="Taipei 101")
  - "The nearest one is at Taipei 101"
- Not enough information? Make a request (location)
  - "Where is your location?"
- Not quite sure? Confirm (type="taiwanese")
  - "Do you want Taiwanese food?"

NLG in dialogue systems

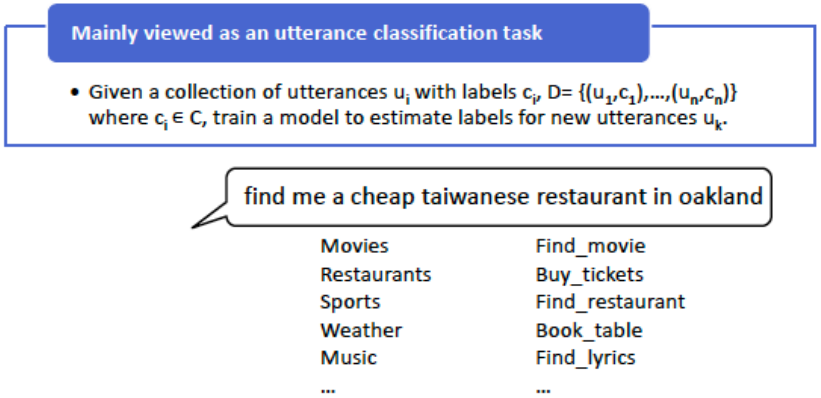
- NLG: generate natural language or GUI given the selected dialogue action for interactions
- Enough information to generate an output utterance (location=“Taipei 101”)
  - “The nearest one is at Taipei 101”

- Not enough information? Make a request (location)
  - “Where is your location?”

- Not quite sure? Confirm (type=“taiwanese”)
  - “Do you want Taiwanese food?”

Implementation examples (1)

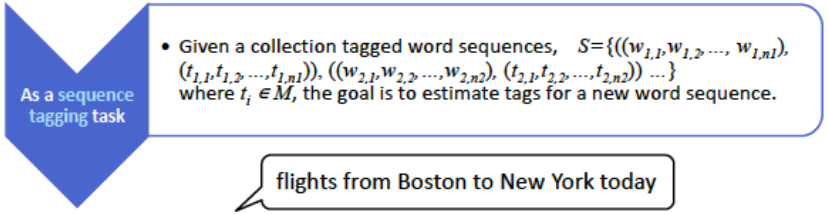
- Classification tasks: domain classification and intent detection



- Examples
  - Deep belief nets, deep convex nets, RNNs, CNNs, large language models (LLMs)

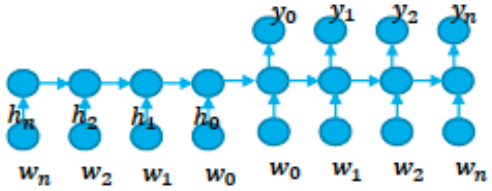
Implementation examples (2)

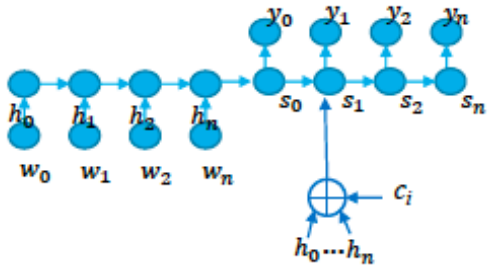
- Sequence tagging tasks (e.g., BIO tagging): slot filling



	flights	from	Boston	to	New	York	today
Entity Tag	O	O	B-city	O	B-city	I-city	O
Slot Tag	O	O	B-dept	O	B-arrival	I-arrival	B-date

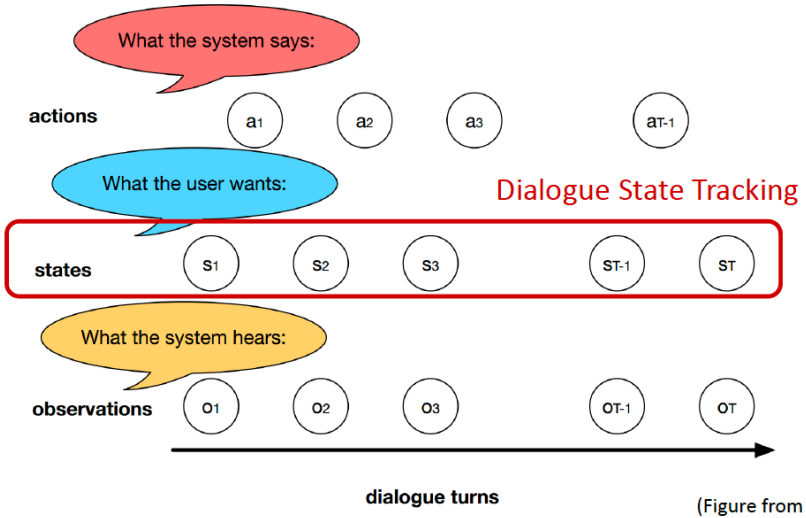
- Examples
  - RNNs, attentions





Implementation examples (3)

- Dialogue state tracking: maintain a belief of the dialogue state and update according to observations
- Examples: neural belief tracker, CNN/RNN based tracker



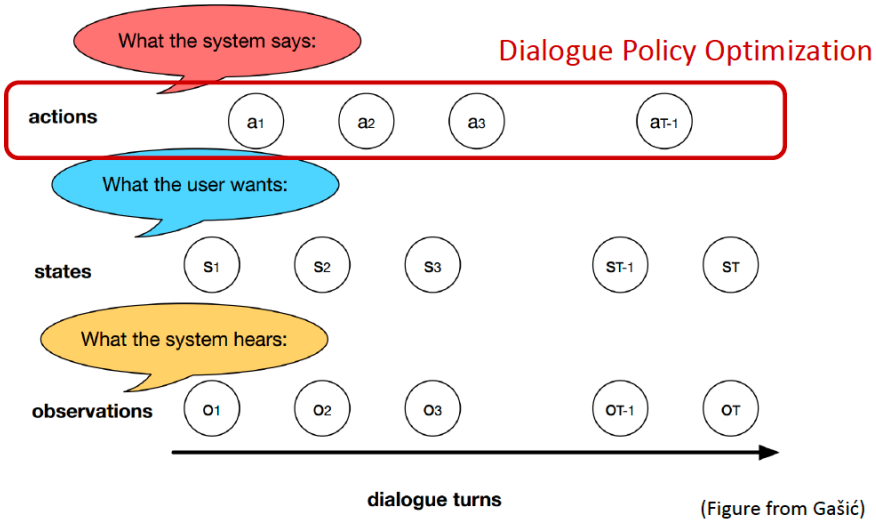
Slot	Value
# people	5 (0.5)
time	5 (0.5)

Slot	Value
# people	3 (0.8)
time	5 (0.8)

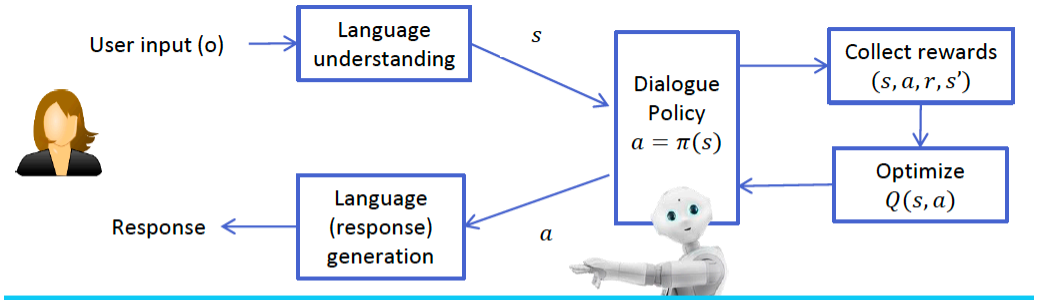
(Figure from Gašić)

Implementation examples (4)

- Dialogue policy learning (DPL): (deep) reinforcement learning



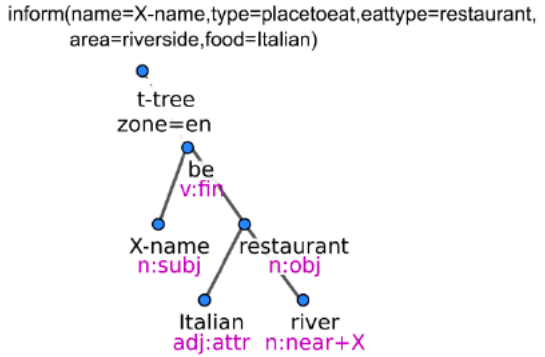
(Figure from Gašić)



Type of Bots	State	Action	Reward
Social ChatBots	Chat history	System Response	# of turns maximized; Intrinsically motivated reward
InfoBots (interactive Q/A)	User current question + Context	Answers to current question	Relevance of answer; # of turns minimized
Task-Completion Bots	User current input + Context	System dialogue act w/ slot value (or API calls)	Task success rate; # of turns minimized

Implementation examples (5)

- NLG
  - seq2seq
  - controlled generation: template based, syntax tree based



Mapping semantic frame into natural language

inform(name=Seven_Days, foodtype=Chinese)

↓

Seven Days is a nice Chinese restaurant

Semantic Frame	Natural Language
confirm()	"Please tell me more about the product your are looking for."
confirm(area=$V)	"Do you want somewhere in the $V?"
confirm(food=$V)	"Do you want a $V restaurant?"
confirm(food=$V,area=$W)	"Do you want a $V restaurant in the $W."

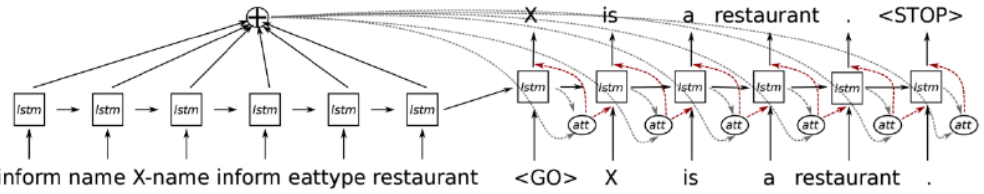
Pros: simple, error-free, easy to control
Cons: time-consuming, poor scalability

(<root> <root> ((X-name n:subj) be v:fin ((Italian adj:attr) restaurant n:obj (river n:near+X))))

X-name n:subj be v:fin Italian adj:attr restaurant n:obj river n:near+X

↓

X is an Italian restaurant near the river.



Implementation examples (6)

- Pretrained models
  - BERT, GPT, DialoGPT pretrained models can be used for various tasks in dialogue systems
  - DialogBERT, Gu et. al., https://arxiv.org/pdf/2012.01775.pdf

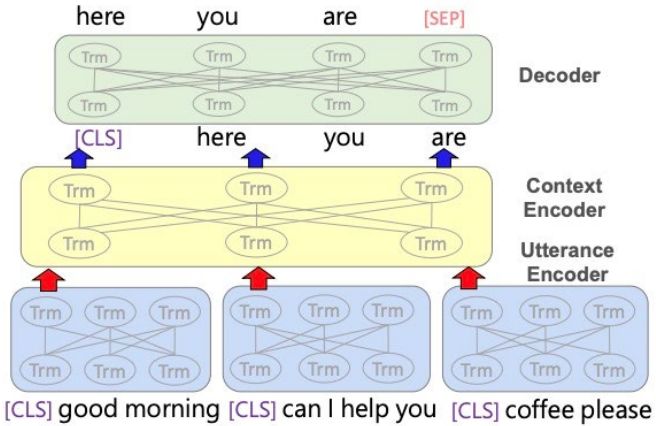


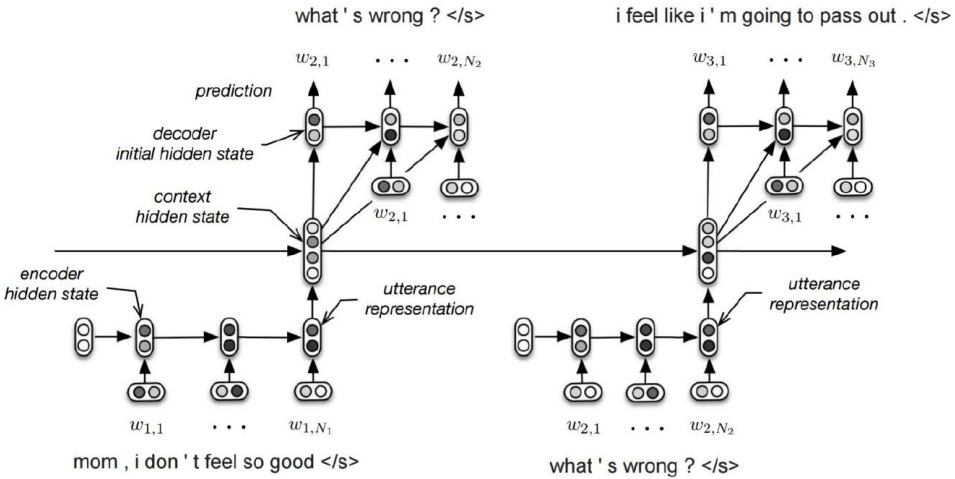
Figure 1: The hierarchical Transformer encoder-decoder architecture. We omit the [SEP] token in the end of each utterance for simplicity.

Two tasks:

1. Masked context regression
2. Distributed utterance order ranking

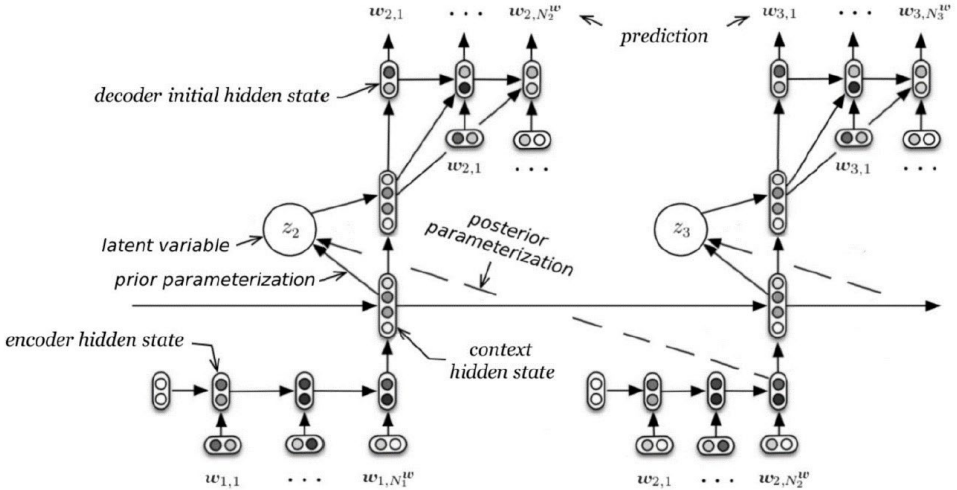
Chitchat dialogue systems

- Learns to generate dialogues from offline dialogue corpora. No state, action, intent, slot, etc.
  - External information/characteristics: sentiment, topic, knowledge ground, etc.
  - E.g., seq2seq, Serban et al., 2016, Building end-to-end dialogue systems using generative hierarchical neural network models



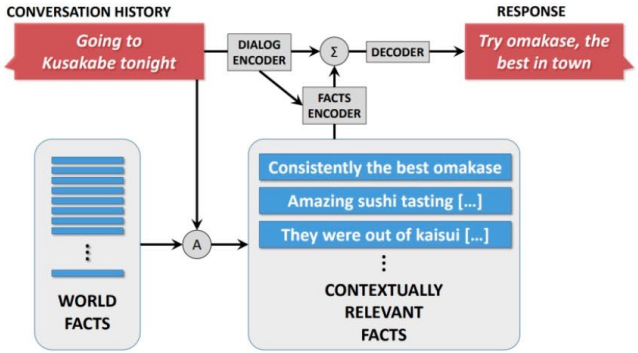
Chitchat dialogue systems with latent variable

- A hierarchical seq2seq model with Gaussian latent variable for generating dialogues (like topic or sentiment)
- Serban et al., 2017, A hierarchical latent variable encoder-decoder model for generating dialogues



Chitchat dialogue systems with knowledge ground

- Ghazvininejadet al., 2017, A knowledge-grounded neural conversation model



A: **Looking forward to trying** *@pizzalibretto* **tonight! my expectations are high**.
B: **Get** *the rocco salad*. **Can you eat** *calamari*?

A: **Anyone in** *Chi* **have a** *dentist office* **they recommend? I'm never going back to** *[...]* **and would love a reco!**
B: **Really looved** *Ora* **in** *Wicker Park*.

A: **I'm at** *California Academy of Sciences*
B: **Make sure you catch** *the show* **at** *the Planetarium*. **Tickets are usually limited.**

A: **I'm at** *New Wave Cafe*.
B: **Try to get to** *Dmitri's* **for dinner**. **Their** *pan fried scallops* **and** *shrimp scampi* **are to die for.**

A: **I just bought:** *[...] 4.3-inch portable GPS navigator* **for my wife, shh, don't tell her**.
B: **I heard this brand** *loses battery power*.

Evaluation

- Ideally want human evaluation, but it's not feasible for training, expensive even for evaluation
- Heuristics: perplexity, BLEU, dialogue length/diversity, fluency, engagingness, and consistency
- Still an open question

Human evaluation on seq2seq model

- 200 questions are picked , and four different humans are asked to rate the seq2seq model (NCM) versus CleverBot (CB).
- Human judges were asked to pick which of the two bots they preferred, and were also allowed to declare a tie if both answers were of equal quality.
- Some samples of evaluation:

Human: What is the deepest spot on the world ?
NCM: The pacific ocean.
CB: What ?

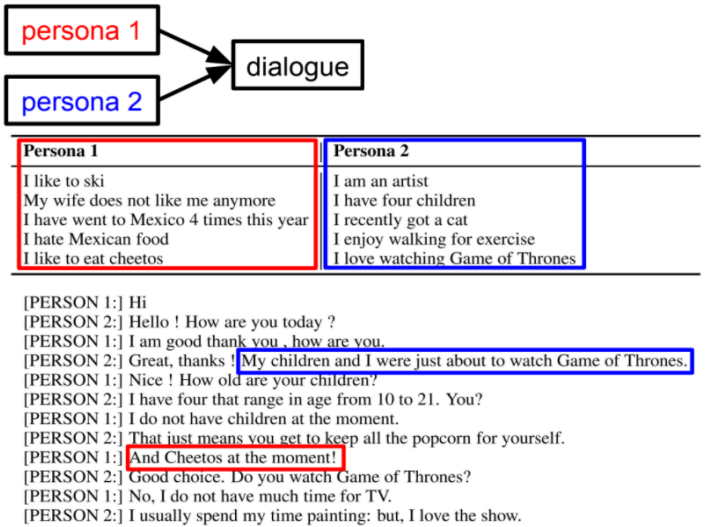
Human: Do you like Mexican food or Indian food ?
NCM: I hate it.
CB: I love Mexican food.

Human: Who are you crazy about ?
NCM: You're the one who's crazy.
CB: Who are you ?

- Evaluation result:
  - NCM model was preferred in 97 out of 200 questions
  - CleverBot was picked in 60 out of 200.
  - There was a tie in 20 questions, and in 23 questions the judges were in disagreement.

Personalizing dialogue agents: PERSONACHAT

- Zhang et al., 2018. Personalizing Dialogue Agents: I have a dog, do you have pets too?
  - Dataset with consistent personalities & evaluate different models
  - Endow each agent with explicit persona



- 1155 personas
- 10,981 dialogues (~19 dialogs per persona)
- 164,356 utterances (sentences)
- 3–5 persona sentences per dialog
- 6–8 chat turns per dialog

Evaluation Metrics

- Perplexity
- Hit@1 accuracy among 20 candidate utterances
- F1 score
- Human evaluation
- Models
  - Ranking models: select response from training set
    - tf-idf BoW based IR baseline
    - StarSpace Embedding [Wu et al., 2017]
    - Ranking Profile Memory Network
    - Key-Value (KV) Profile Memory Network
  - Generative models: generate word by word
    - Seq2Seq
    - Generative Profile Memory Network

Evaluation: ranking model

Method	No Persona		Self Persona		Their Persona		Both Personas	
	Orig	Rewrite	Orig	Rewrite	Orig	Rewrite	Orig	Rewrite
IR baseline	0.214	0.214	0.410	0.207	0.181	0.181	0.382	0.188
Training on original personas								
Starspace	0.318	0.318	0.481	0.295	0.245	0.235	0.429	0.258
Profile Memory	0.318	0.318	0.473	0.302	0.283	0.267	0.438	0.266
Training on revised personas								
Starspace	0.318	0.318	0.491	0.322	0.271	0.261	0.432	0.288
Profile Memory	0.318	0.318	0.509	0.354	0.299	0.294	0.467	0.331
KV Profile Memory	0.349	0.349	0.511	0.351	0.291	0.289	0.467	0.330

Table 6: **Evaluation of dialog utterance prediction with ranking models** using hits@1 in four settings: conditioned on the speakers persona ("self persona"), the dialogue partner's persona ("their persona"), both or none. The personas are either the original source given to Turkers to condition the dialogue, or the rewritten personas that do not have word overlap, explaining the poor performance of IR in that case.

Evaluation: generative models

Persona	Method	Original ppl	Original hits@1	Original F1	Revised ppl	Revised hits@1	Revised F1
No Persona		38.08	0.092	0.168	38.08	0.092	0.168
Self Persona	Seq2Seq	40.53	0.084	**0.172**	40.65	0.082	**0.171**
	Profile Memory	**34.54**	**0.125**	**0.172**	38.21	**0.108**	0.170
Their Persona	Seq2Seq	41.48	0.075	0.168	41.95	0.074	0.168
	Profile Memory	36.42	0.105	0.167	**37.75**	0.103	0.167
Both Personas	Seq2Seq	40.14	0.084	0.169	40.53	0.082	0.166
	Profile Memory	35.27	0.115	0.171	38.48	0.106	0.168

Table 5: **Evaluation of dialog utterance prediction with generative models** in four settings: conditioned on the speakers persona ("self persona"), the dialogue partner's persona ("their persona"), both or none. The personas are either the original source given to Turkers to condition the dialogue, or the revised personas that do not have word overlap. In the "no persona" setting, the models are equivalent, so we only report once.

Human evaluation

Method Model	Profile	Fluency	Engagingness	Consistency	Persona Detection
Human	Self	4.31(1.07)	4.25(1.06)	4.36(0.92)	0.95(0.22)
Generative PersonaChat Models					
Seq2Seq	None	3.17(1.10)	3.18(1.41)	2.98(1.45)	0.51(0.50)
Profile Memory	Self	3.08(1.40)	3.13(1.39)	3.14(1.26)	0.72(0.45)
Ranking PersonaChat Models					
KV Memory	None	3.81(1.14)	3.88(0.98)	3.36(1.37)	0.59(0.49)
KV Profile Memory	Self	3.97(0.94)	3.50(1.17)	3.44(1.30)	0.81(0.39)
Twitter LM	None	3.21(1.54)	1.75(1.04)	1.95(1.22)	0.57(0.50)
OpenSubtitles 2018 LM	None	2.85(1.46)	2.13(1.07)	2.15(1.08)	0.35(0.48)
OpenSubtitles 2009 LM	None	2.25(1.37)	2.12(1.33)	1.96(1.22)	0.38(0.49)
OpenSubtitles 2009 KV Memory	None	2.14(1.20)	2.22(1.22)	2.06(1.29)	0.42(0.49)

Table 4: **Human Evaluation** of various PERSONA-CHAT models, along with a comparison to human performance, and Twitter and OpenSubtitles based models (last 4 rows), standard deviation in parenthesis.

Summary

- Task-oriented dialogue systems
- Chitchat dialogue systems
- Performance evaluation
- Challenges (not covered in the lecture)
  - Knowledge accuracy (Factoid)
  - Ethics
  - Bias
  - Explainability
  - Privacy and data security