Deep Learning Approaches to Text Production

Claire Gardent
CNRS/LORIA, Nancy

Shashi Narayan
University of Edinburgh

NAACL 2018, New Orleans

Laboratoire lorrain de recherche en informatique et ses applications

University of Edinburgh Natural Language Processing

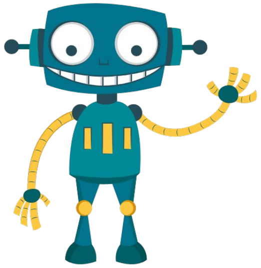
What's the Purpose of this Tutorial?

- Text production and its relevance?
- Why deep learning for text production?

Text Production: From What and What for?

Verbalise
Respond
Summarize
Simplify
...

Communicative Goal



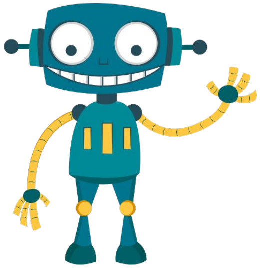
Databases
Knowledge bases
Dialog acts
Sentences
Documents
...

Input

Text Production: From What and What for?

Verbalise
Respond
Summarize
Simplify
...

Communication Goal

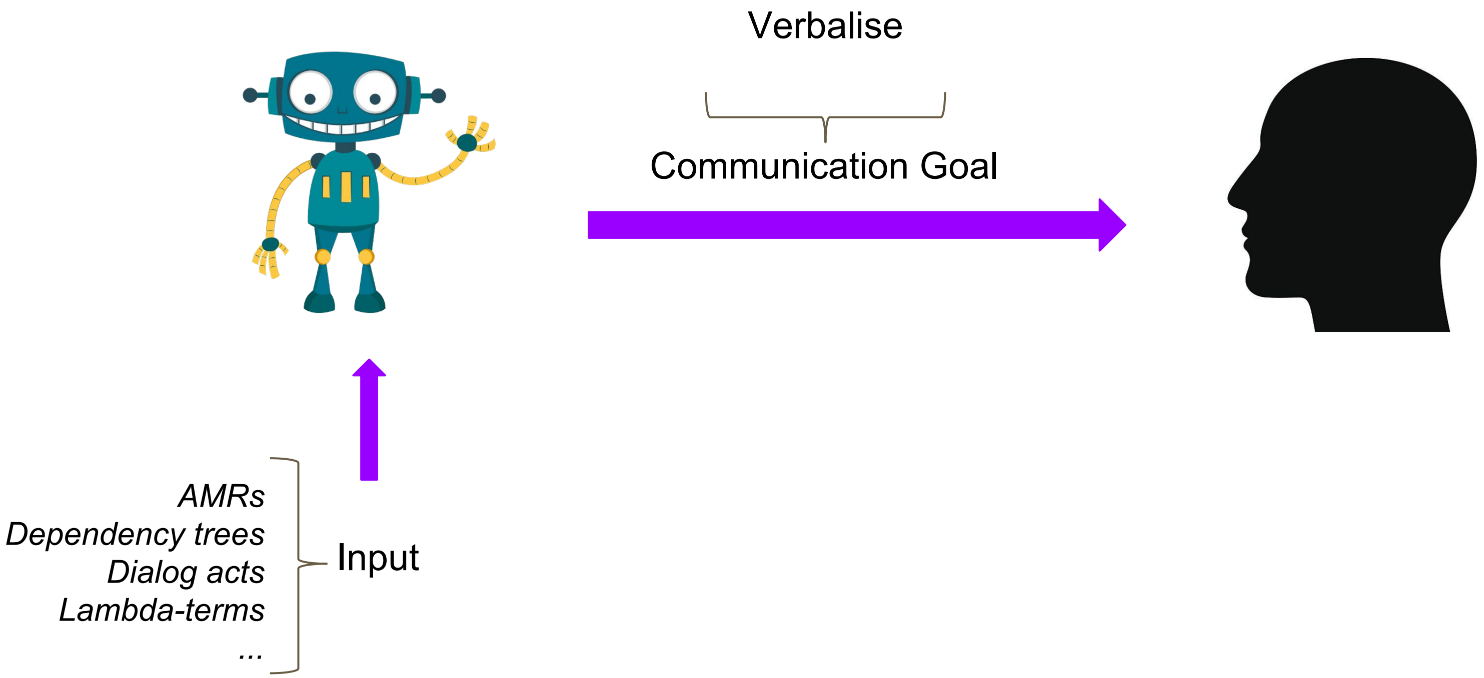


- Meaning representations
- Data
- Text

Databases
Knowledge bases
Dialog acts
Sentences
Documents
...

Input

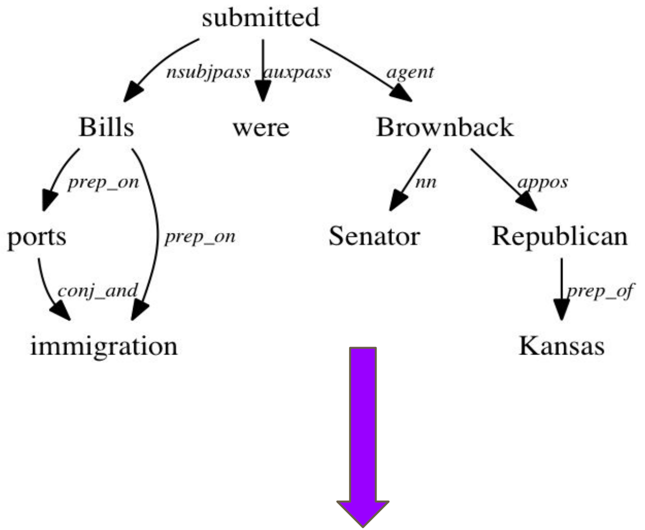
Meaning Representation to Text Production



Generating from Dependency Trees

Surface Realization Challenge 2011 and 2018

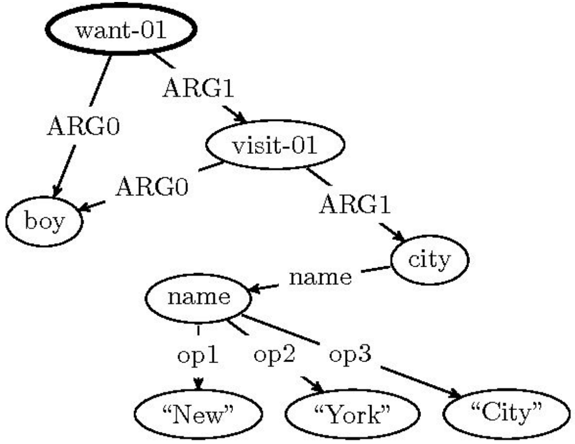
- Shallow and deep approaches
- Universal dependency trees



Bills on immigration were submitted by Senator Brownback, a Republican of Kansas.

Generating from Abstract Meaning Representations

SemEval Shared Task 2017: AMR Generation and Parsing



(a) Graph.

```
(w / want-01
  :ARG0 (b / boy)
  :ARG1 (g / visit-01
           :ARG0 b
           :ARG1 (c / city
                    :name (n / name
                             :op1 "New"
                             :op2 "York"
                             :op3 "City"))))
```

(b) AMR annotation.

A boy wants to visit New York City.
A boy wanted to visit New York City.

Generating from Dialog Moves

Input:

name[The Eagle],
eatType[coffee shop],
food[French],
priceRange[moderate],
customerRating[3/5],
area[riverside],
kidsFriendly[yes],
near[Burger King]

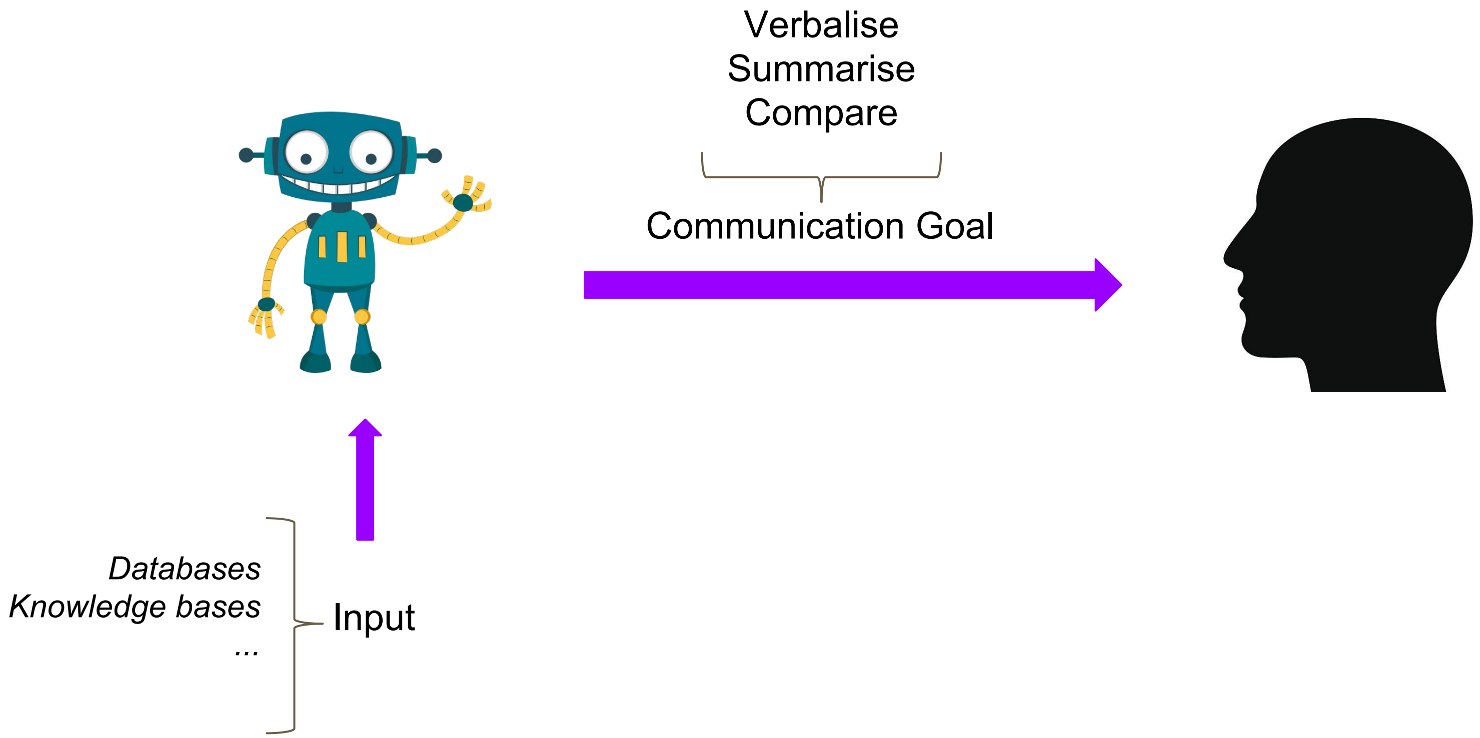
Mairesse and Young 2014

Wen et al. 2015, 2016

End-to-End Natural Language Generation Challenge 2017

Output: *"The three star coffee shop, The Eagle, gives families a mid-priced dining experience featuring a variety of wines and cheeses. Find The Eagle near Burger King."*

Data to Text Production



Generating from Knowledge Bases (RDFs)

The WebNLG Challenge 2017

(John_E_Blaha *birthDate* 1942_08_26) (John_E_Blaha *birthPlace* San_Antonio) (John_E_Blaha *occupation* Fighter_pilot)

"John E Blaha, born in San Antonio on 1942-08-26, worked as a fighter pilot."

Generating from Databases

(a)

Flight	
From	To
phoenix	new_york

Search	
Type	What
query	flight

Day	
Day	Dep/Ar
sunday	departure

List flights from phoenix to new york on sunday

(b)

Temperature			
Time	Min	Mean	Max
06:00-21:00	9	15	21

Cloud Sky Cover	
Time	Percent (%)
06:00-09:00	25-50
09:00-12:00	50-75

Wind Speed			
Time	Min	Mean	Max
06:00-21:00	15	20	30

Wind Direction	
Time	Mode
06:00-21:00	S

Cloudy, with a low around 10. South wind around 20 mph.

(c)

Pass	
From	To
pink3	pink7

Bad Pass	
From	To
pink7	purple3

Turn Over	
From	To
pink7	purple3

pink3 passes the ball to pink7

Angeli, Liang and Klein 2010. Konstas and Lapata 2012

Generating from Data to Documents

TEAM	WIN	LOSS	PTS	FG_PCT	RB	AS ...
Heat	11	12	103	49	47	27
Hawks	7	15	95	43	33	20

PLAYER	AS	RB	PT	FG	FGA	CITY ...
Tyler Johnson	5	2	27	8	16	Miami
Dwight Howard	4	17	23	9	11	Atlanta
Paul Millsap	2	9	21	8	12	Atlanta
Goran Dragic	4	2	21	8	17	Miami
Wayne Ellington	2	3	19	7	15	Miami
Dennis Schroder	7	4	17	8	15	Atlanta
Rodney McGruder	5	5	11	3	8	Miami
Thabo Sefolosha	5	5	10	5	11	Atlanta
Kyle Korver	5	3	9	3	9	Atlanta
...						

The Atlanta Hawks defeated the Miami Heat , 103 - 95 , at Philips Arena on Wednesday . Atlanta was in desperate need of a win and they were able to take care of a shorthanded Miami team here . Defense was key for the Hawks , as they held the Heat to 42 percent shooting and forced them to commit 16 turnovers . Atlanta also dominated in the paint , winning the rebounding battle , 47 - 34 , and outscoring them in the paint 58 - 26.The Hawks shot 49 percent from the field and assisted on 27 of their 43 made baskets . This was a near wire - to - wire win for the Hawks , as Miami held just one lead in the first five minutes . Miami (7 - 15) are as beat - up as anyone right now and it 's taking a toll on the heavily used starters . Hassan Whiteside really struggled in this game , as he amassed eight points , 12 rebounds and one blocks on 4 - of - 12 shooting ...

Wiseman, Shieber and Rush 2017

Generating from Loosely Aligned Data

Born	Robert Joseph Flaherty February 16, 1884 Iron Mountain, Michigan, U.S.
Died	July 23, 1951 (aged 67) Dummerston, Vermont, U.S.
Cause of death	Cerebral thrombosis
Occupation	Filmmaker
Spouse(s)	Frances Johnson Hubbard

Robert Joseph Flaherty, (**February 16, 1884 July 23, 1951**) was an **American film-maker**. Flaherty was married to **Frances H. Flaherty** until his death in 1951.

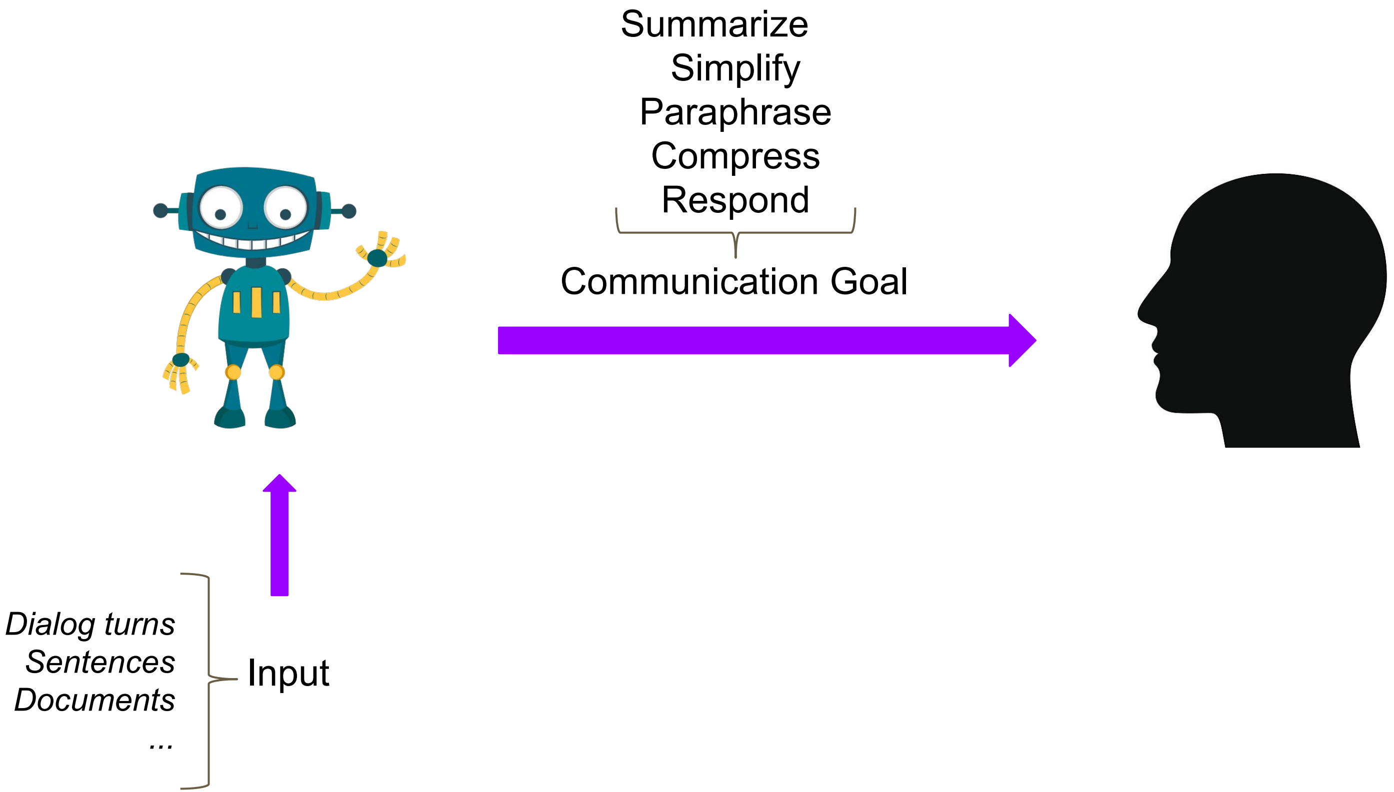
Perez and Lapata, NAACL 2018

Other Data Modalities to Text Production

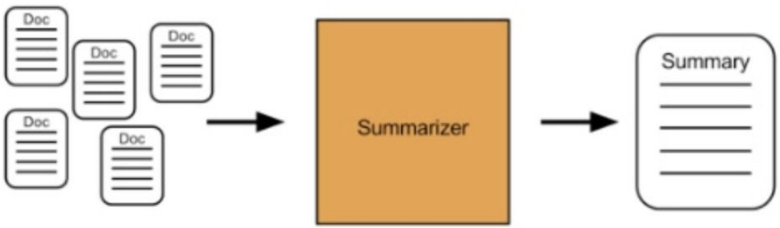
Generating Image Captions

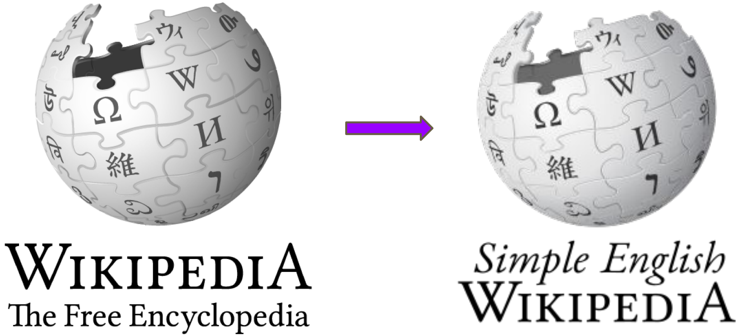
Vinyals et al. 2015

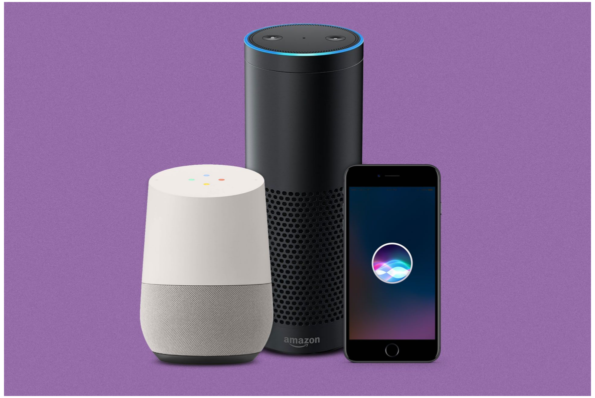
Text to Text Production



Text to Text Production







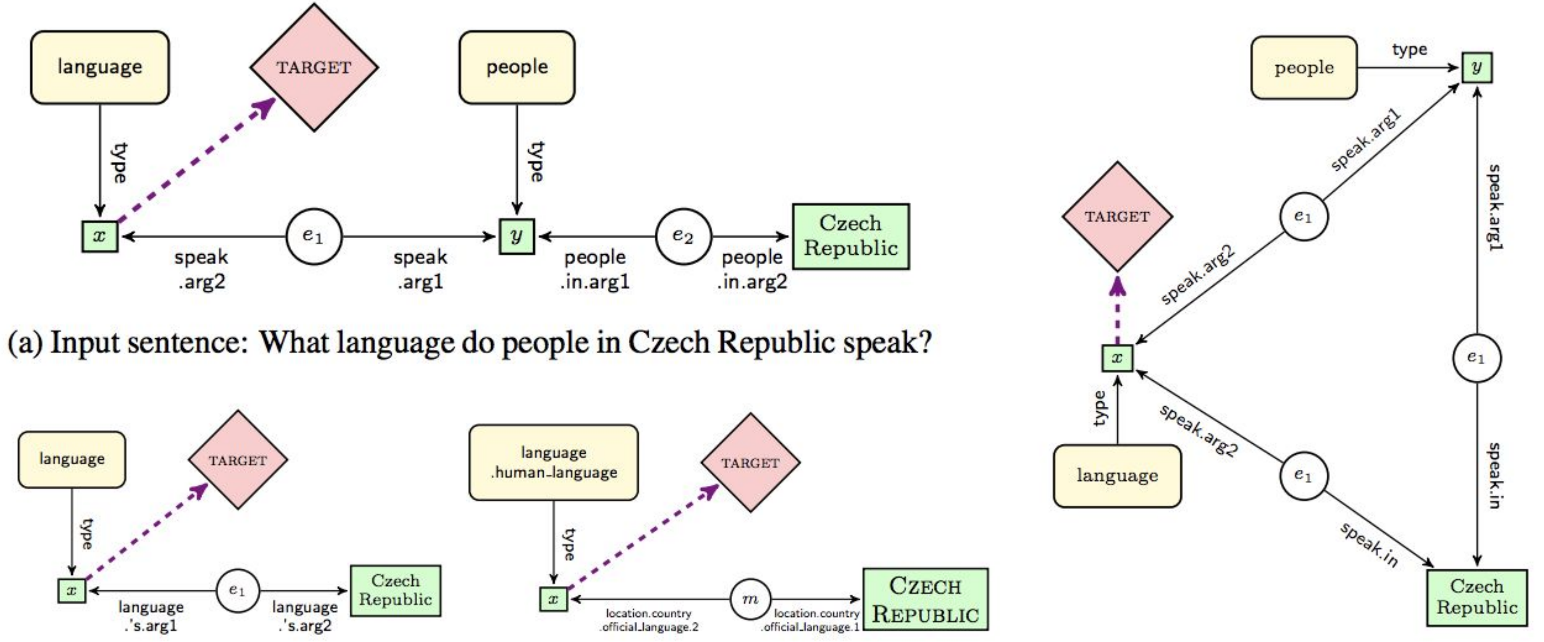
Sentence Simplification

Simple English
WIKIPEDIA

Complex: In 1964 Peter Higgs published his second paper in Physical Review Letters describing Higgs mechanism which predicted a new massive spin-zero boson for the first time.

Simple: Peter Higgs wrote his paper explaining Higgs mechanism in 1964. Higgs mechanism predicted a new elementary particle.

Paraphrasing and Question Answering



(a) Input sentence: What language do people in Czech Republic speak?

(c) Paraphrase: What is Czech Republic's language?

(d) Freebase grounded graph

(b) Paraphrase: What language do people speak in Czech Republic?

Abstract Generation

REVIEW

doi:10.1038/nature14539

Deep learning

Yann LeCun[1,2], Yoshua Bengio[3] & Geoffrey Hinton[4,5]

Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction. These methods have dramatically improved the state-of-the-art in speech recognition, visual object recognition, object detection and many other domains such as drug discovery and genomics. Deep learning discovers intricate structure in large data sets by using the backpropagation algorithm to indicate how a machine should change its internal parameters that are used to compute the representation in each layer from the representation in the previous layer. Deep convolutional nets have brought about breakthroughs in processing images, video, speech and audio, whereas recurrent nets have shone light on sequential data such as text and speech.

Machine-learning technology powers many aspects of modern society: from web searches to content filtering on social networks to recommendations on e-commerce websites, and it is increasingly present in consumer products such as cameras and smartphones. Machine-learning systems are used to identify objects in images, transcribe speech into text, match news items, posts or products with users' interests, and select relevant results of search. Increasingly, these applications make use of a class of techniques called deep learning.

Conventional machine-learning techniques were limited in their ability to process natural data in their raw form. For decades, constructing a pattern-recognition or machine-learning system required careful engineering and considerable domain expertise to design a feature extractor that transformed the raw data (such as the pixel values of an image) into a suitable internal representation or feature vector from which the learning subsystem, often a classifier, could detect or classify patterns in the input.

intricate structures in high-dimensional data and is therefore applicable to many domains of science, business and government. In addition to beating records in image recognition[1–4] and speech recognition[5–7], it has beaten other machine-learning techniques at predicting the activity of potential drug molecules[8], analysing particle accelerator data[9,10], reconstructing brain circuits[11], and predicting the effects of mutations in non-coding DNA on gene expression and disease[12,13]. Perhaps more surprisingly, deep learning has produced extremely promising results for various tasks in natural language understanding[14], particularly topic classification, sentiment analysis, question answering[15] and language translation[16,17].

We think that deep learning will have many more successes in the near future because it requires very little engineering by hand, so it can easily take advantage of increases in the amount of available computation and data. New learning algorithms and architectures that are currently being developed for deep neural networks will only accelerate this progress.

Headline or Title Generation

Story Highlights Generation

Did Cambridge Analytica get YOUR data? Facebook will tell 87 million affected users TODAY if their information was shared

- **Starting today all 2.2 billion Facebook users will receive a notice on their feeds**
- **'Protecting Your Information,' notice will contain link to see what apps they use**
- **This will also show what information they have shared with those apps, firm says**
- **Users whose data may have been shared with Cambridge Analytica will be told**
- **Facebook says most of the 87 million affected users are in the United States**
- **In an interview Sunday, Cambridge Analytica whistleblower Christopher Wylie said the number could actually be larger than 87 million**

By ASSOCIATED PRESS

PUBLISHED: 01:14, 9 April 2018 | **UPDATED:** 08:58, 9 April 2018

Share 930 shares 158 View comments

Today Facebook will tell the 87 million users who may have had their information shared with Cambridge Analytica.

Starting Monday all 2.2 billion **Facebook** users will receive a notice on their newsfeeds, titled 'Protecting Your Information,' with a link to see what apps they use and what information they have shared with those apps.

If they want, they can shut off apps individually or turn off third-party access to their apps completely.

Multi-document Summarization

India backs down on proposed "fake news" legislation after an outcry

- On Tuesday, the Indian government walked back a new rule that would have punished publishers of so-called "fake news," after many questioned what exactly would fall into that category. [CNN / Sugam Pokharel and Joshua Berlinger]
- On Monday, the government announced that journalists who were found to have written "fake news" would lose their official accreditation, in some cases permanently. But the proposal faced such swift and strong backlash that by Tuesday, the government had changed its tune. [NYT / Kai Schultz and Suhasini Raj]
- Many in the Indian news media saw the new rules as an attack on the press, noting organizations like the Press Council of India and the News Broadcasters Association already exist to ensure press accountability. [Times of India]
- Indian journalists also pointed out that the amendment was released mere months before campaigning was set to begin for national elections in 2019, and that Prime Minister Narendra Modi's party has a history of attacking members of the press who publish criticism of their leadership. [Times of India]

Conversational Agents

A: Where are you going? (1)
B: I'm going to the police station. (2)
A: I'll come with you. (3)
B: No, no, no, no, you're not going anywhere. (4)
A: Why? (5)
B: I need you to stay here. (6)
A: I don't know what you are talking about. (7)
...

Li, Monroe, Ritter, Galley, Gao and Jurafsky 2016

Summary

- Many different inputs

  Data, Meaning Representations, Text

- Many different communicative goals

  Verbalise, summarise, compress, simplify, respond, compare ....

Pre-neural Approaches

Previous Approaches to Text Production

Data-to-Text Generation

Text Planning | Sentence Planning

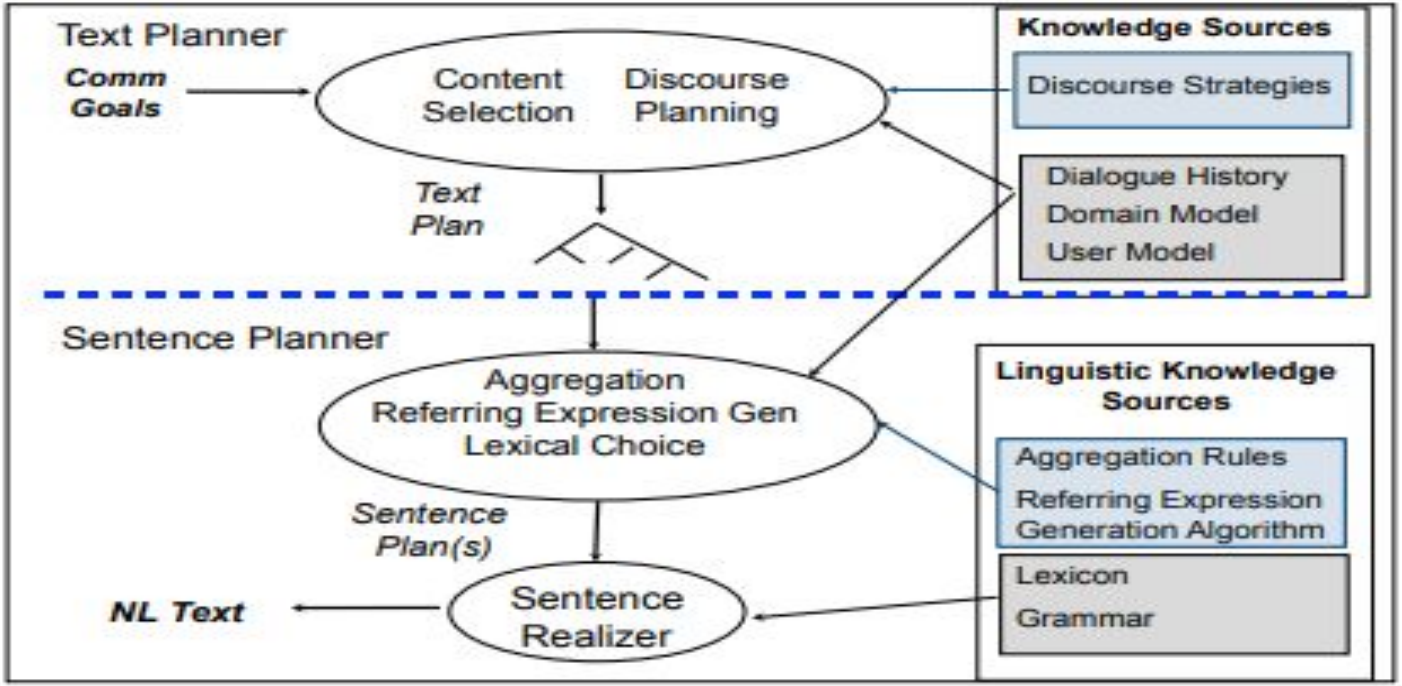
Simplification, Compression and Paraphrasing

Split | Rewrite | Reorder | Delete

Summarisation

Content Selection | Aggregation | Generalisation

The Data-to-Text Generation Pipeline



(Figure from Johanna Moore)

The Data-to-Text Generation Pipeline

Pros

Models the various choices which need to be made during Generation

Cons

- Many modules to implement
- Error propagation
- Difficult to capture the interactions between the various choices (joint learning)

Simplification, Compression and Paraphrasing

Four main operations

- Delete
- Reorder
- Rewrite
- Split

Simplification, Compression and Paraphrasing

In 1964 Peter Higgs published his ~~second~~ paper ~~in Physical Review Letters~~ describing Higgs mechanism [which] predicted a new massive spin-zero boson ~~for the first time~~.

REORDER

Peter Higgs wrote his paper explaining Higgs mechanism in 1964.

Higgs mechanism predicted a new elementary particle.

Simplification, Compression and Paraphrasing

In 1964 Peter Higgs published his ~~second~~ paper ~~in Physical Review Letters~~ describing Higgs mechanism [which] predicted a new massive spin-zero boson ~~for the first time~~.

SPLIT

Peter Higgs wrote his paper explaining Higgs mechanism in 1964.

Higgs mechanism predicted a new elementary particle.

Simplification, Compression and Paraphrasing

In 1964 Peter Higgs published his ~~second~~ paper ~~in Physical Review Letters~~ describing Higgs mechanism [which] predicted a new massive spin-zero boson ~~for the first time~~.

DELETE

Peter Higgs wrote his paper explaining Higgs mechanism in 1964.

Higgs mechanism predicted a new elementary particle.

Simplification, Compression and Paraphrasing

In 1964 Peter Higgs published his ~~second~~ paper ~~in Physical Review Letters~~ describing Higgs mechanism [which] predicted a new massive spin-zero boson ~~for the first time~~.

REWRITE

Peter Higgs wrote his paper explaining Higgs mechanism in 1964.

Higgs mechanism predicted a new elementary particle.

Simplification, Compression and Paraphrasing

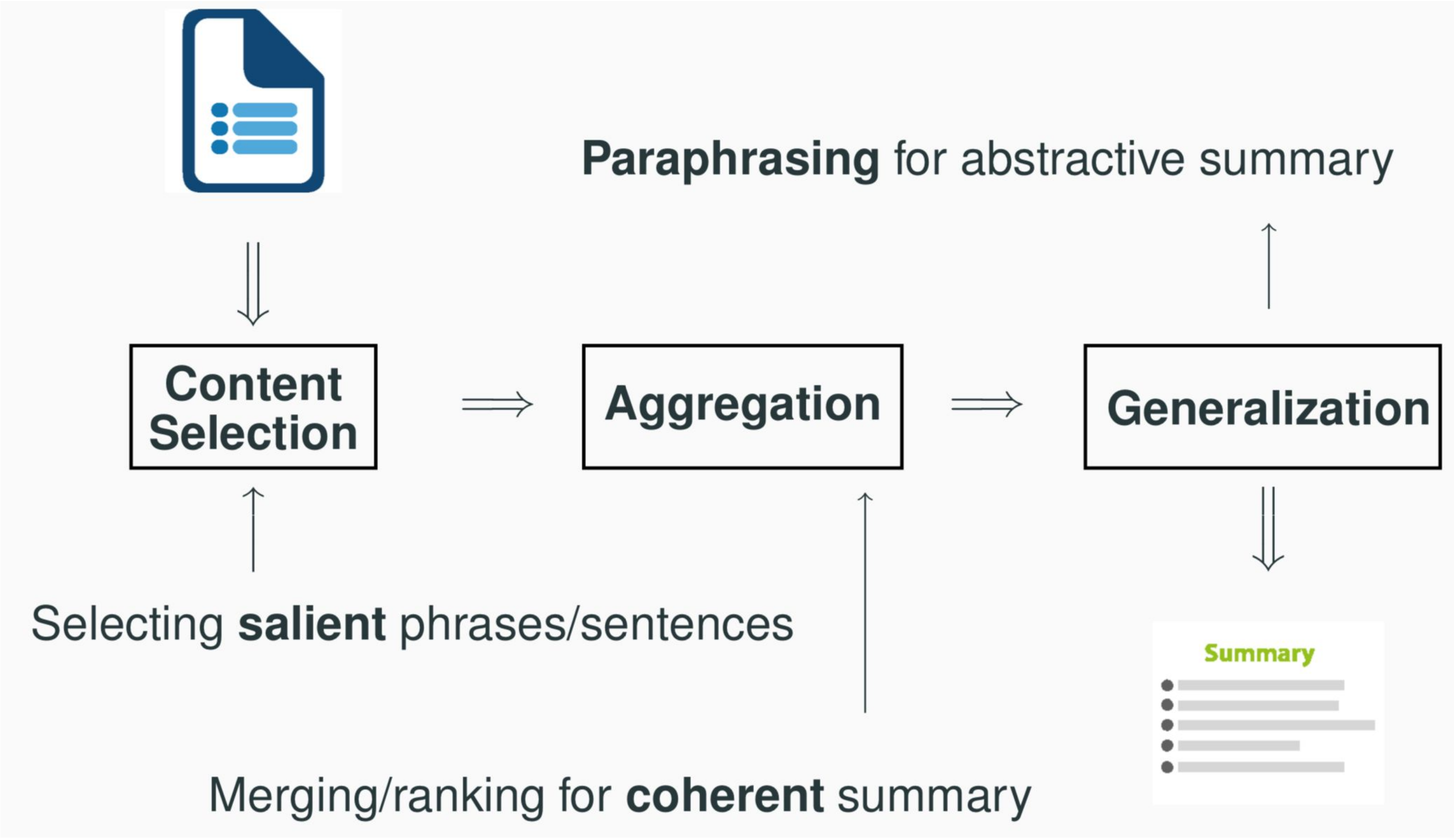
Pros

- Fine grained control over the four operations

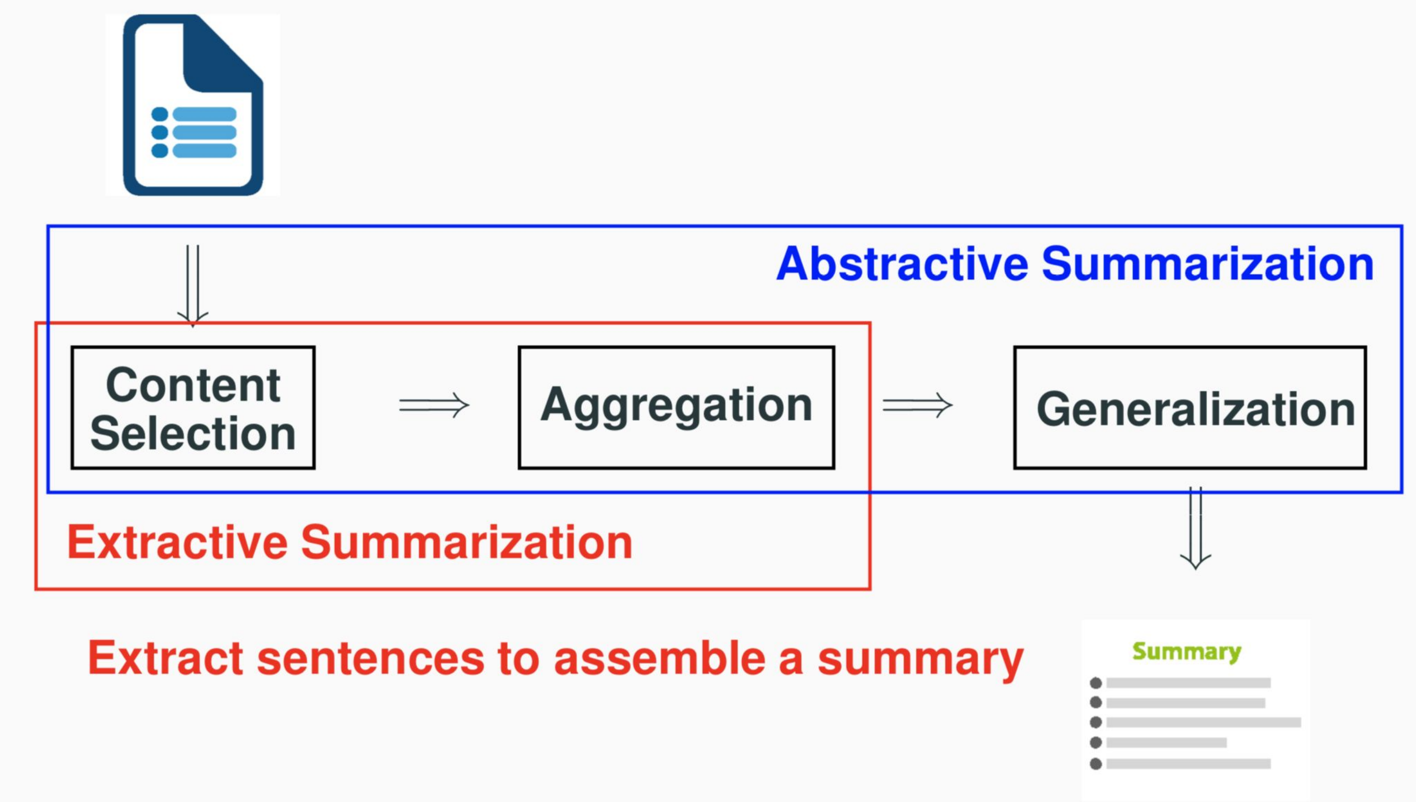
Cons

- Hard to capture the interactions between operations

Summarization



Summarization



Summarization

Pros

- Well-formed (grammatical) summaries
- Fast
- Reasonable performance

Cons

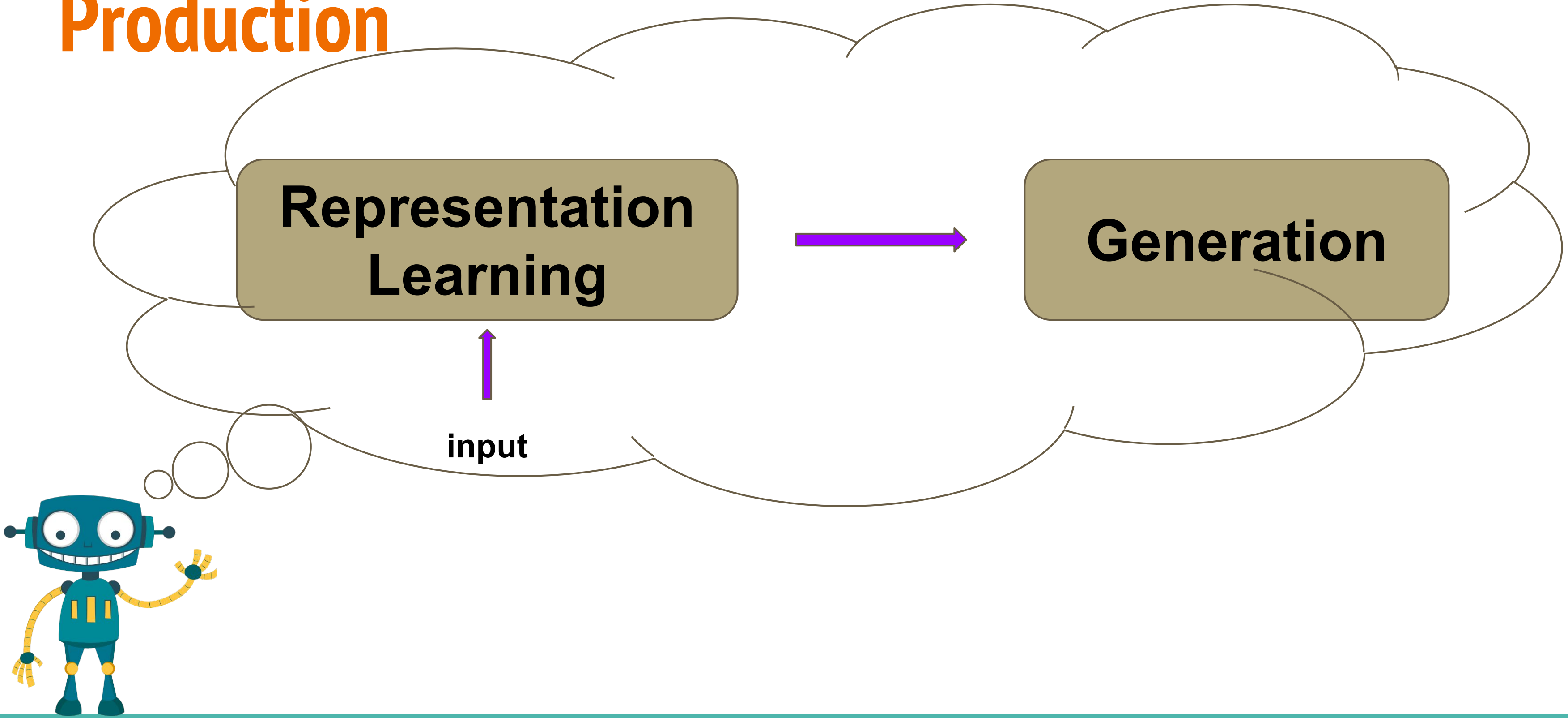
- Extractive summaries are still very different from human written summaries

Questions ?

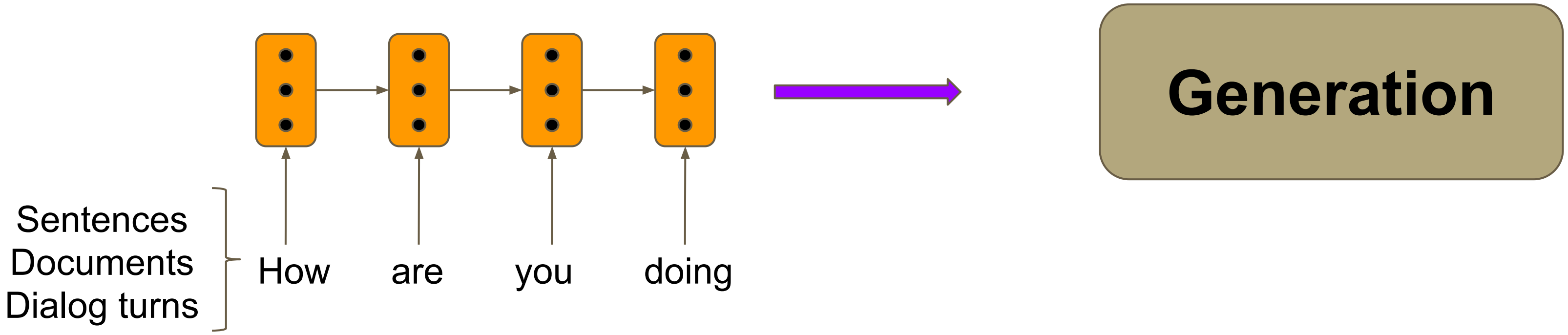
Neural Text Production

- A single framework for all text production tasks
- End-to-end

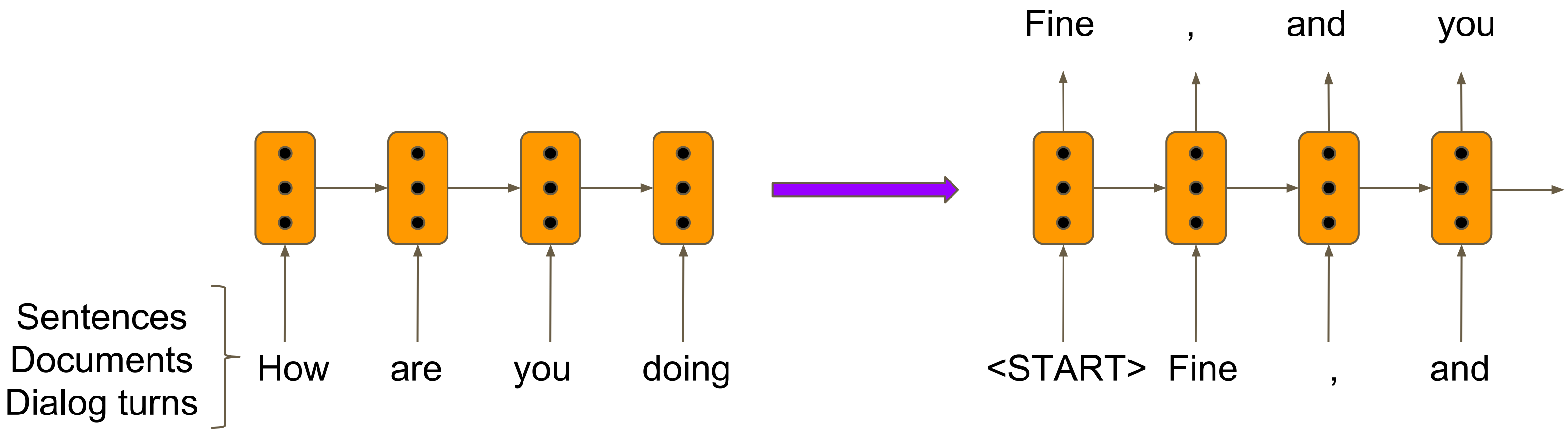
Deep Learning: A Uniform Framework for Text Production



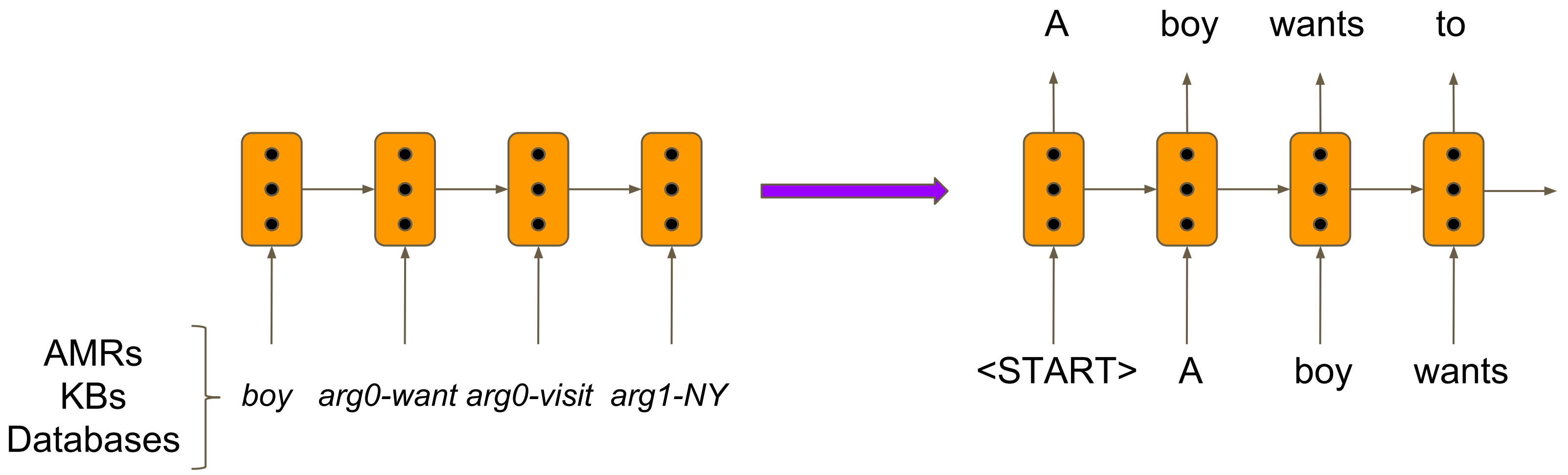
Deep Learning: A Uniform Framework for Text Production



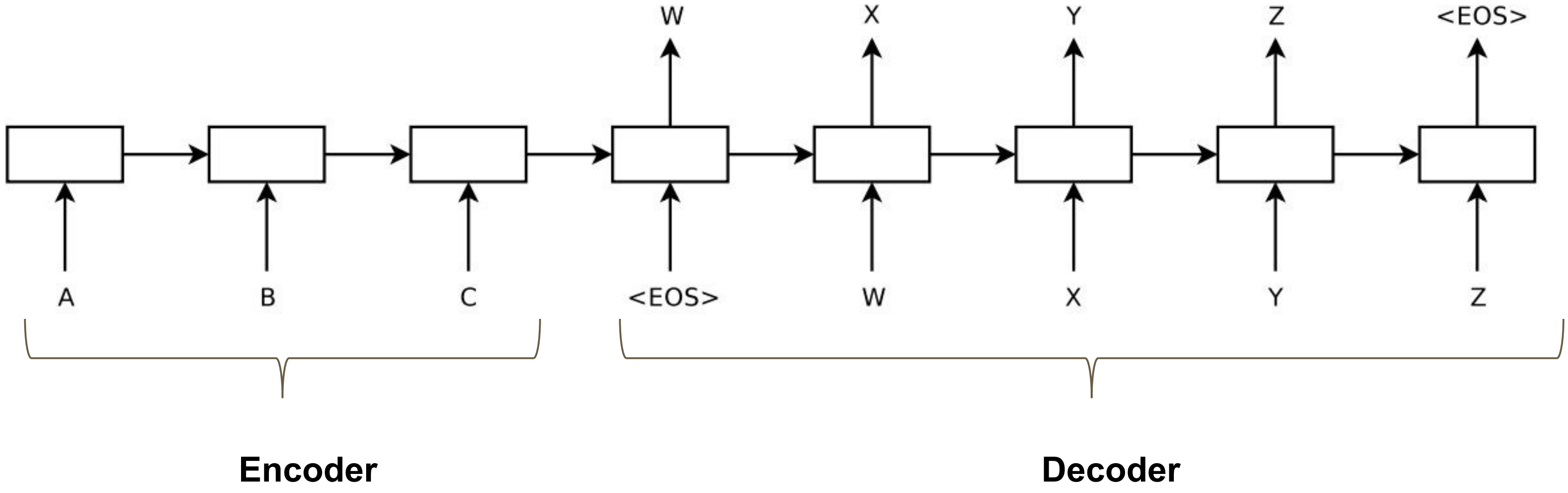
Deep Learning: A Uniform Framework for Text Production



Deep Learning: A Uniform Framework for Text Production

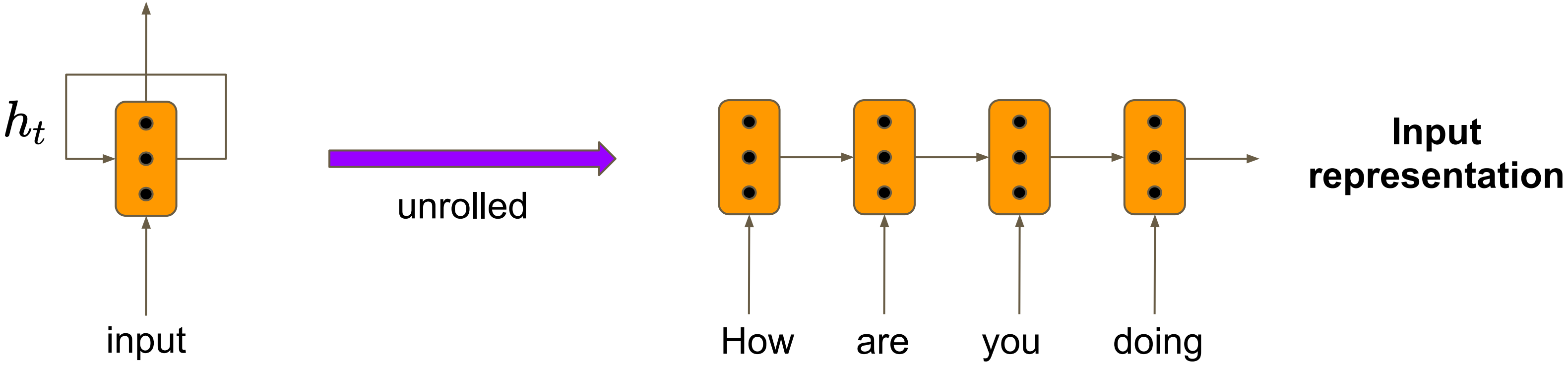


Encoder-Decoder Model for Text Production



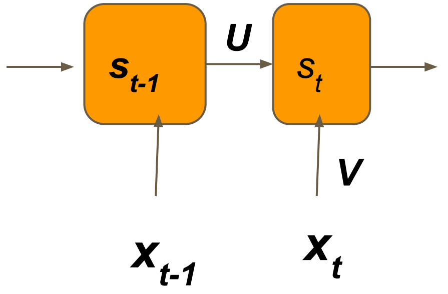
Cho et al. 2014, Sutskever et al. 2014

Encoding Input Representations using Recurrent Neural Networks (RNN)



Encoding variable length inputs

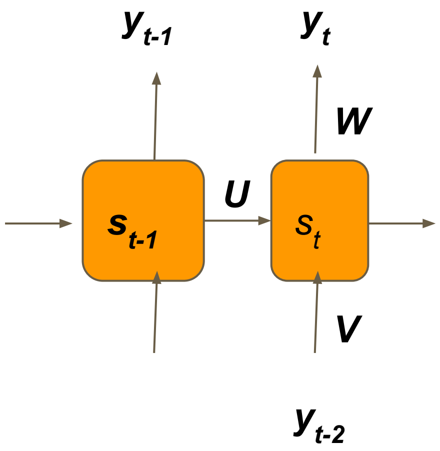
Encoding Representations using an RNN



$$s_t = tanh(U * s_{t-1} + V * x_t)$$

The decoder takes as input the previous state $\boldsymbol{s_{t-1}}$ and the previously generated token $\boldsymbol{y_{t-1}}$

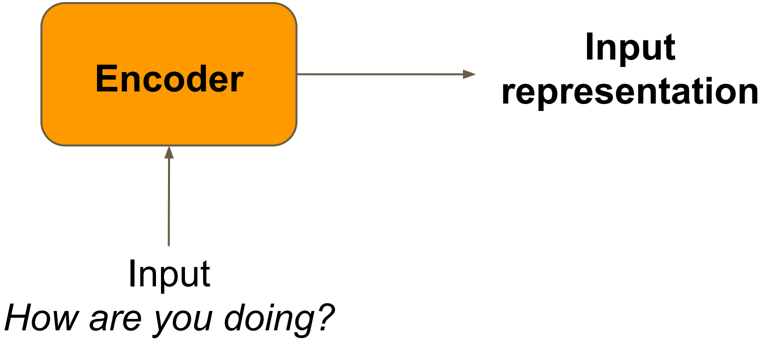
Decoding Representations using an RNN



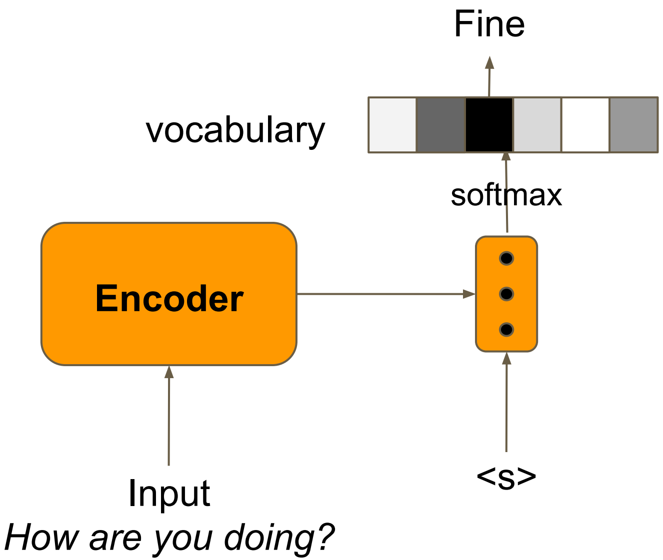
$$
\begin{aligned}
s_t &= tanh(U * s_{t-1} + V * y_{t-1}) \\
y_t &= softmax(W * s_t)
\end{aligned}
$$

The decoder takes as input the previous state $\boldsymbol{s_{t-1}}$ and the previously generated token $\boldsymbol{y_{t-1}}$

Generating Text using RNNs



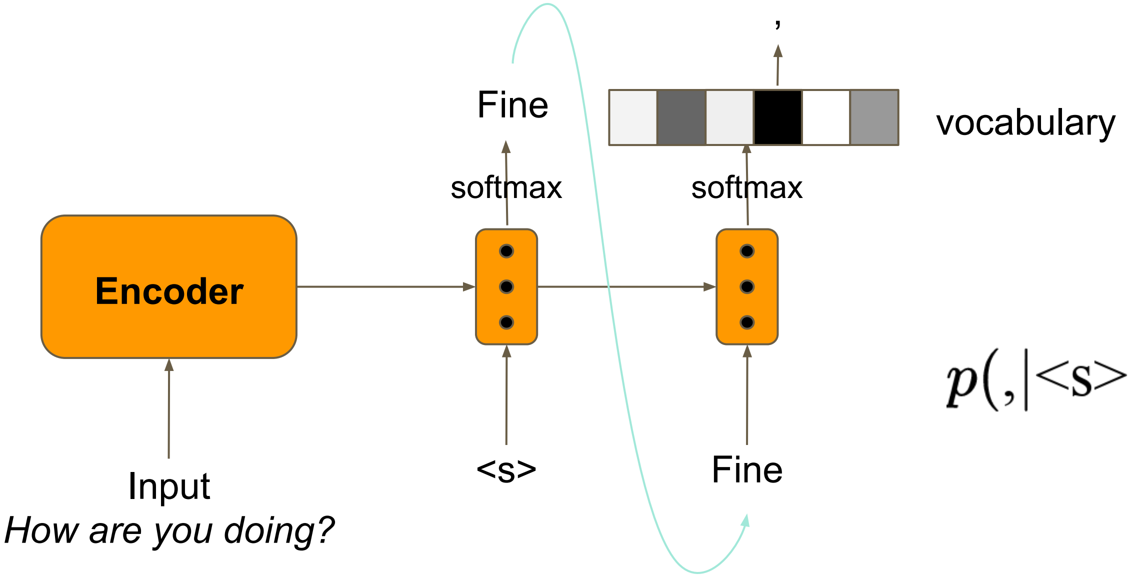
Generating Text using RNNs



$$p(\text{Fine}|\text{<s>}, \text{How are you doing?})$$

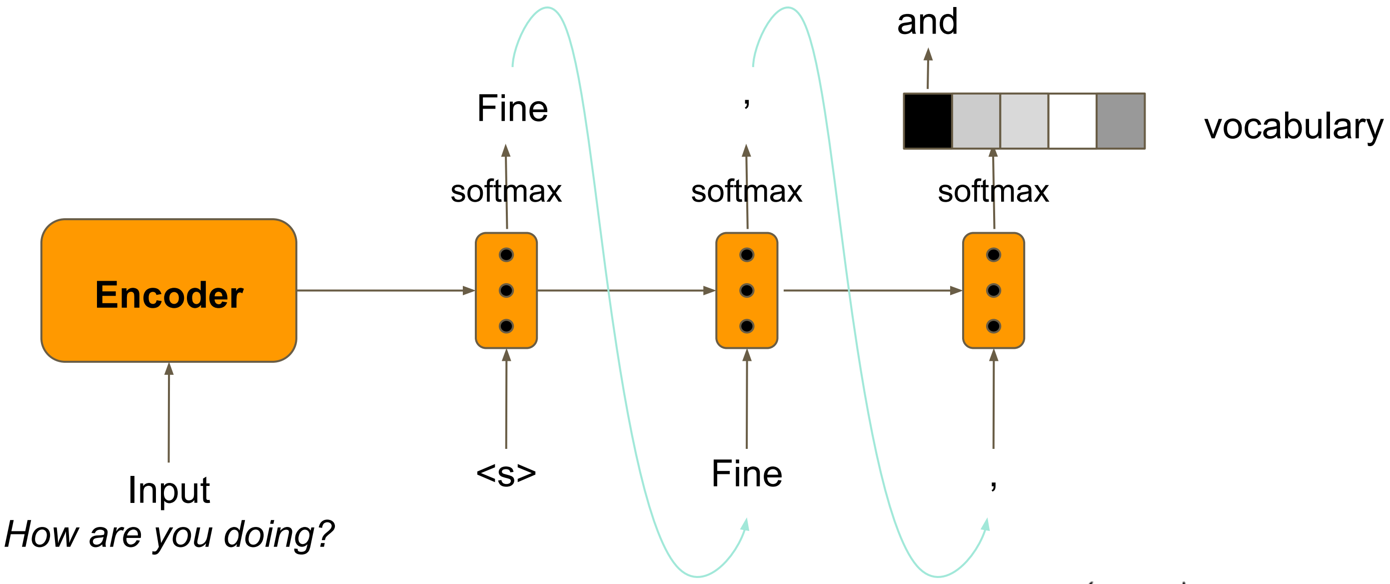
Conditional Generation

Generating Text using RNNs



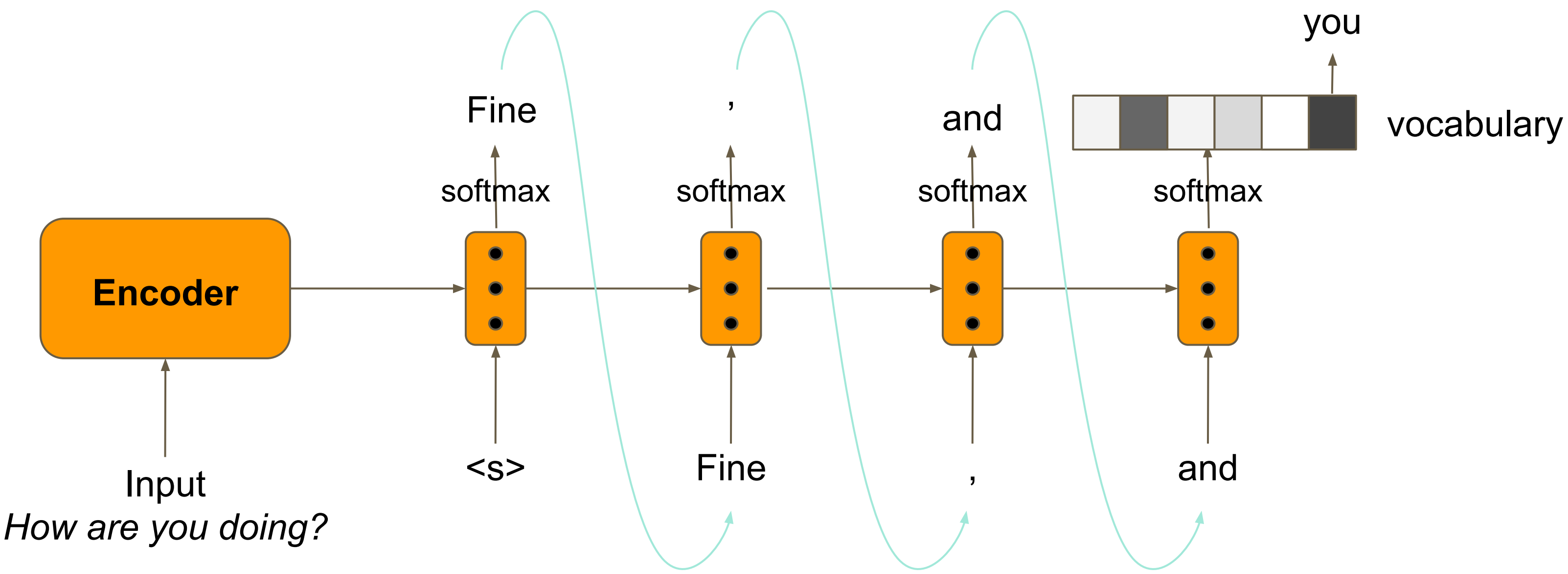
$$p(, | \text{<s> Fine, How are you doing?})$$

Generating Text using RNNs

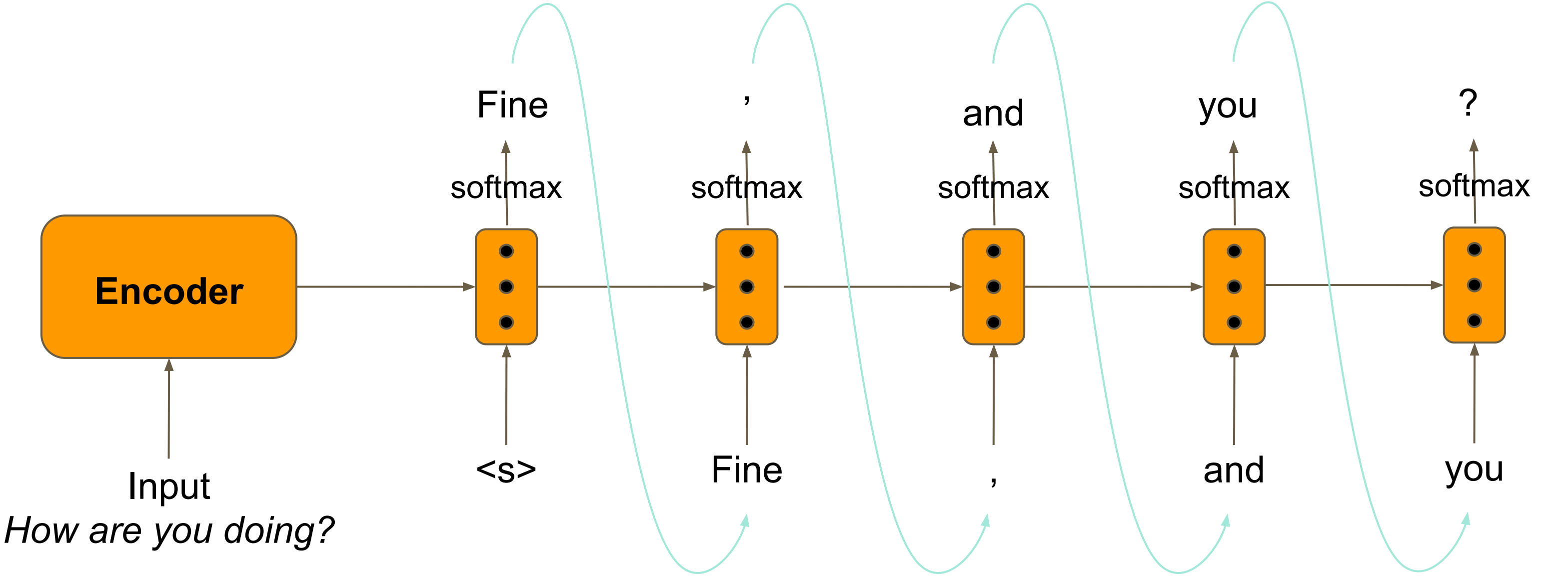


$$p(\text{and} \mid \text{<s> Fine,; How are you doing?})$$

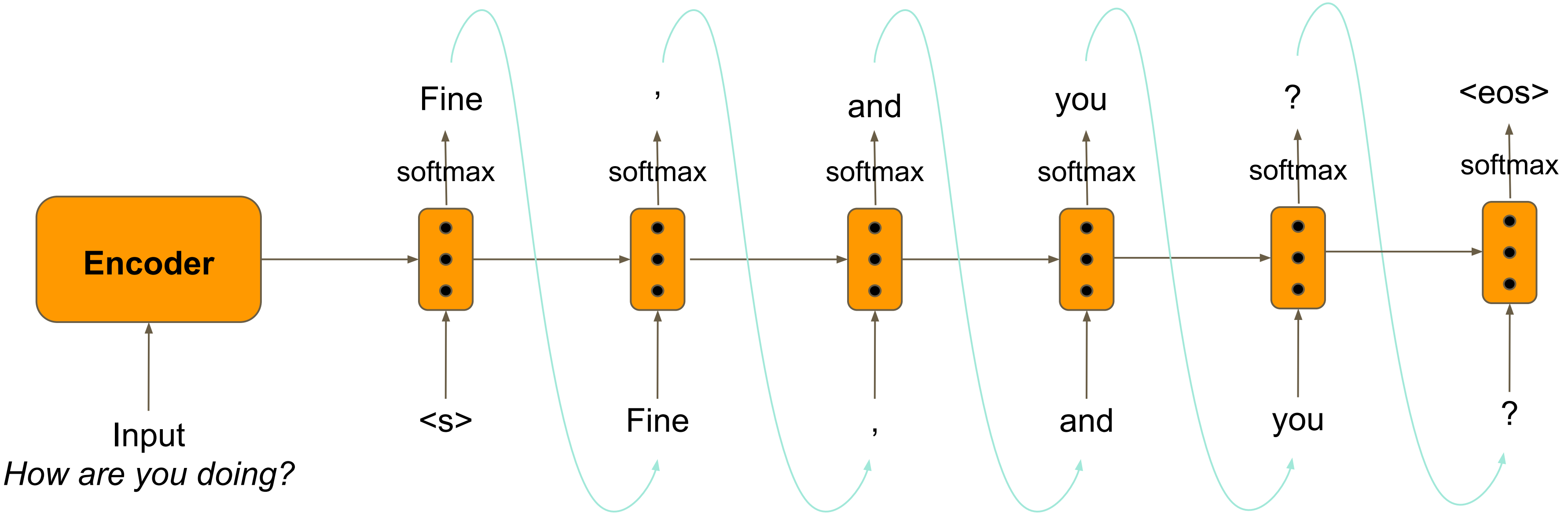
Generating Text using RNNs



Generating Text using RNNs



Generating Text using RNNs



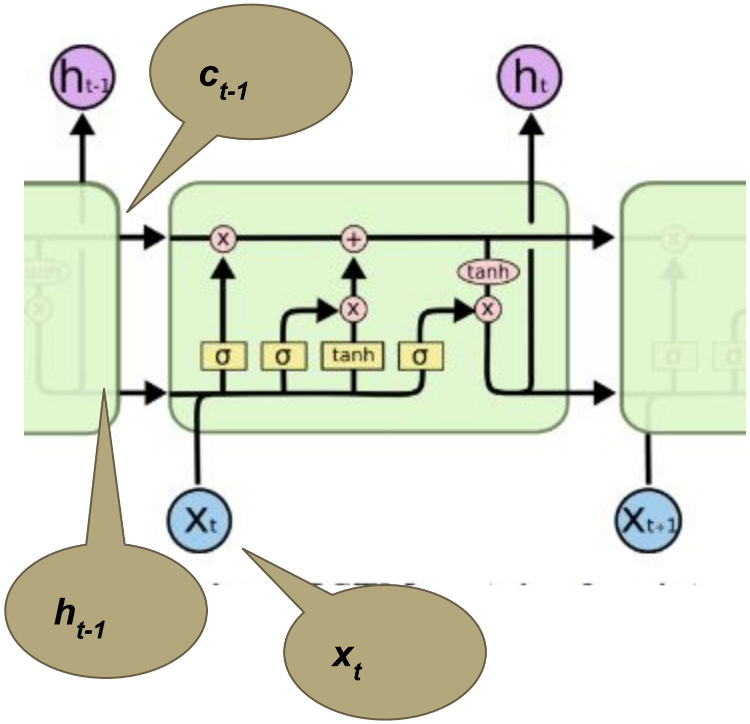
RNN and Long Distance Dependencies

- In practice, RNN cannot handle long input because of the Vanishing and Exploding Gradients issue [Bengio et al. 1994]

The yogi, who had done many sun salutations, was happy.

- LSTM, GRU are alternative recurrent networks which helps learning long distance dependencies

Long Short Term Memory networks (LSTMs)



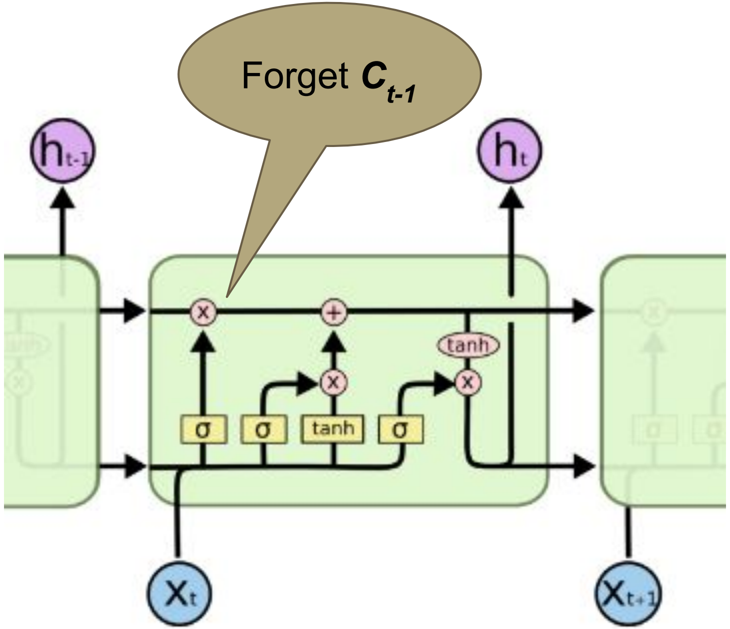
update

forget

output

$$
\begin{aligned}
\tilde{c}_t &= tanh(W_c * [h_{t-1}, x_t]) \\
u_t &= \sigma(W_t * [h_{t-1}, x_t]) \\
f_t &= \sigma(W_f * [h_{t-1}, x_t]) \\
o_t &= \sigma(W_o * [h_{t-1}, x_t]) \\
c_t &= u_t * \tilde{c}_t + f_t * c_{t-1} \\
h_t &= o_t * tanh(c_t)
\end{aligned}
$$

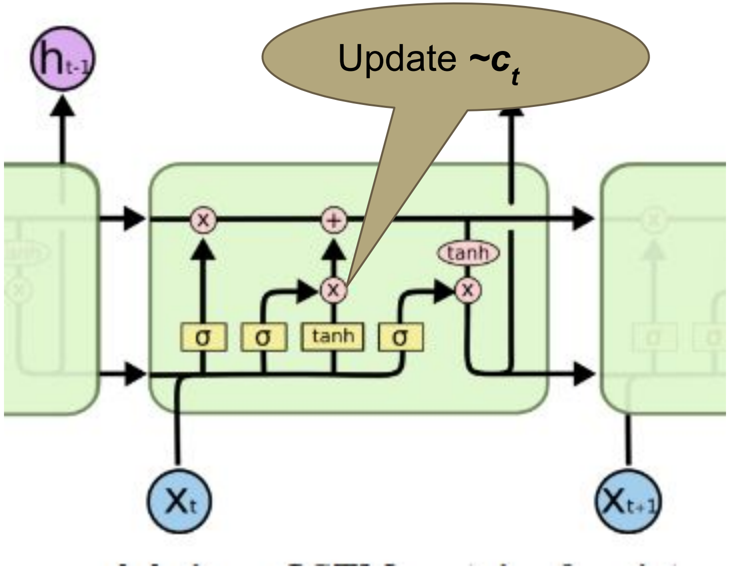
Long Short Term Memory networks (LSTMs)



update
forget
output

$$
\begin{aligned}
\tilde{c}_t &= tanh(W_c * [h_{t-1}, x_t]) \\
u_t &= \sigma(W_t * [h_{t-1}, x_t]) \\
f_t &= \sigma(W_f * [h_{t-1}, x_t]) \\
o_t &= \sigma(W_o * [h_{t-1}, x_t]) \\
c_t &= u_t * \tilde{c}_t + f_t * c_{t-1} \\
h_t &= o_t * tanh(c_t)
\end{aligned}
$$

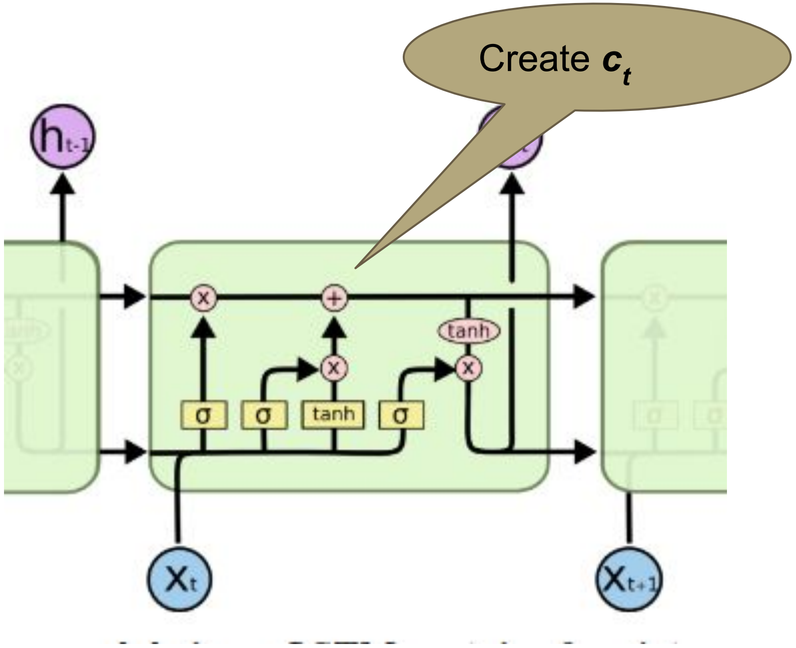
Long Short Term Memory networks (LSTMs)



update
forget
output

$$
\begin{aligned}
\tilde{c}_t &= tanh(W_c * [h_{t-1}, x_t]) \\
u_t &= \sigma(W_t * [h_{t-1}, x_t]) \\
f_t &= \sigma(W_f * [h_{t-1}, x_t]) \\
o_t &= \sigma(W_o * [h_{t-1}, x_t]) \\
c_t &= u_t * \tilde{c}_t + f_t * c_{t-1} \\
h_t &= o_t * tanh(c_t)
\end{aligned}
$$

Long Short Term Memory networks (LSTMs)



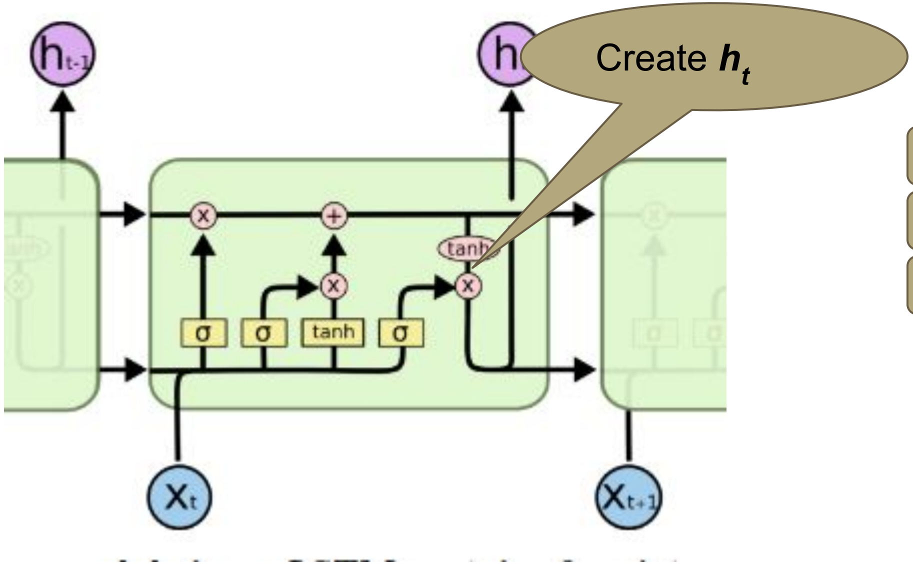
update

forget

output

$$
\begin{aligned}
\tilde{c}_t &= tanh(W_c * [h_{t-1}, x_t]) \\
u_t &= \sigma(W_t * [h_{t-1}, x_t]) \\
f_t &= \sigma(W_f * [h_{t-1}, x_t]) \\
o_t &= \sigma(W_o * [h_{t-1}, x_t]) \\
c_t &= u_t * \tilde{c}_t + f_t * c_{t-1} \\
h_t &= o_t * tanh(c_t)
\end{aligned}
$$

Long Short Term Memory networks (LSTMs)



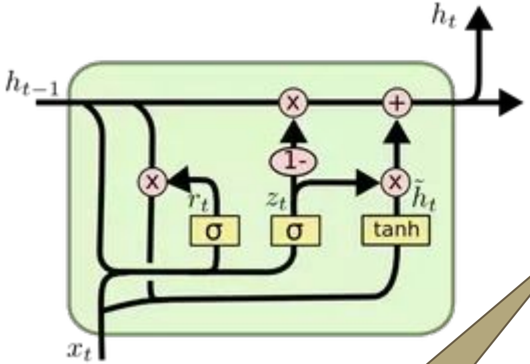
update

forget

output

$$
\begin{aligned}
\tilde{c}_t &= tanh(W_c * [h_{t-1}, x_t]) \\
u_t &= \sigma(W_t * [h_{t-1}, x_t]) \\
f_t &= \sigma(W_f * [h_{t-1}, x_t]) \\
o_t &= \sigma(W_o * [h_{t-1}, x_t]) \\
c_t &= u_t * \tilde{c}_t + f_t * c_{t-1} \\
h_t &= o_t * tanh(c_t)
\end{aligned}
$$

Gated Recurrent Units (GRUs)



$$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$$

$$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$$

$$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

Candidate state

Z_t = 0, output previous state (***memorize***)

Z_t = 1, output candidate state (***forget***)

Bidirectional Recurrent Neural Networks

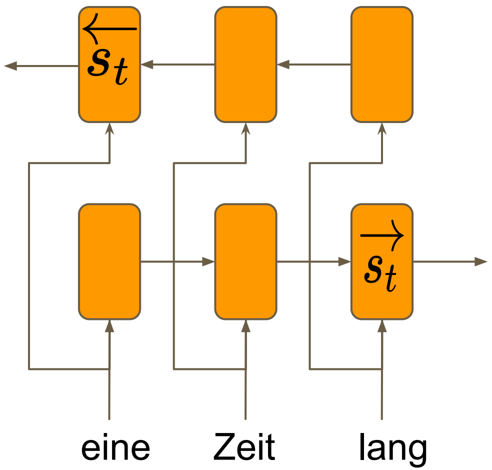
*Ich habe keine **Zeit***

*I don't have **time***

*Er schwieg eine **Zeit** lang*

*He was silent **for a while***

Bidirectional Recurrent Neural Networks



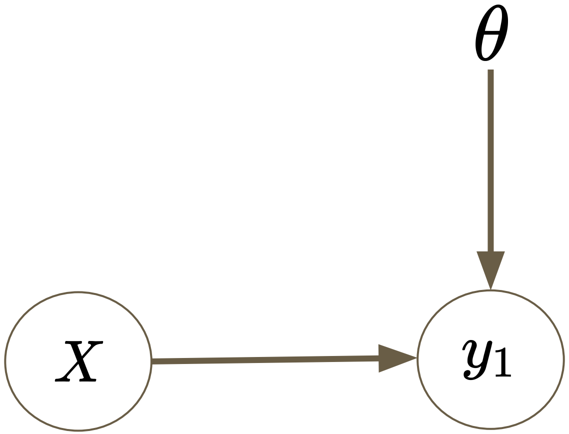
Input representation

$$[\overrightarrow{s_t}\ ,\ \overleftarrow{s_t}]$$

- Forward RNN encodes left context
- Backward RNN encodes right context
- Forward and backward states are concatenated

Summary: Generating Text using RNNs

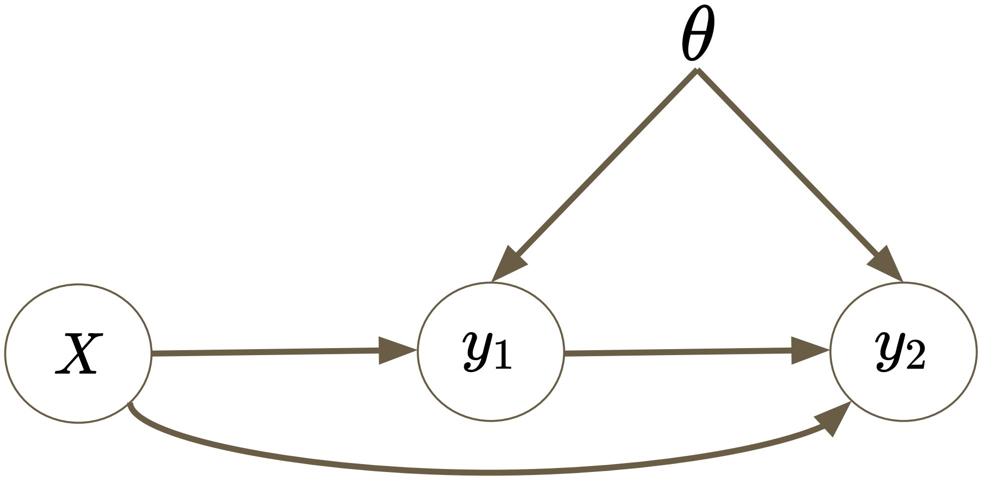
A conditional language model with no markov assumption



$$p(y_1|X;\theta)$$

Summary: Generating Text using RNNs

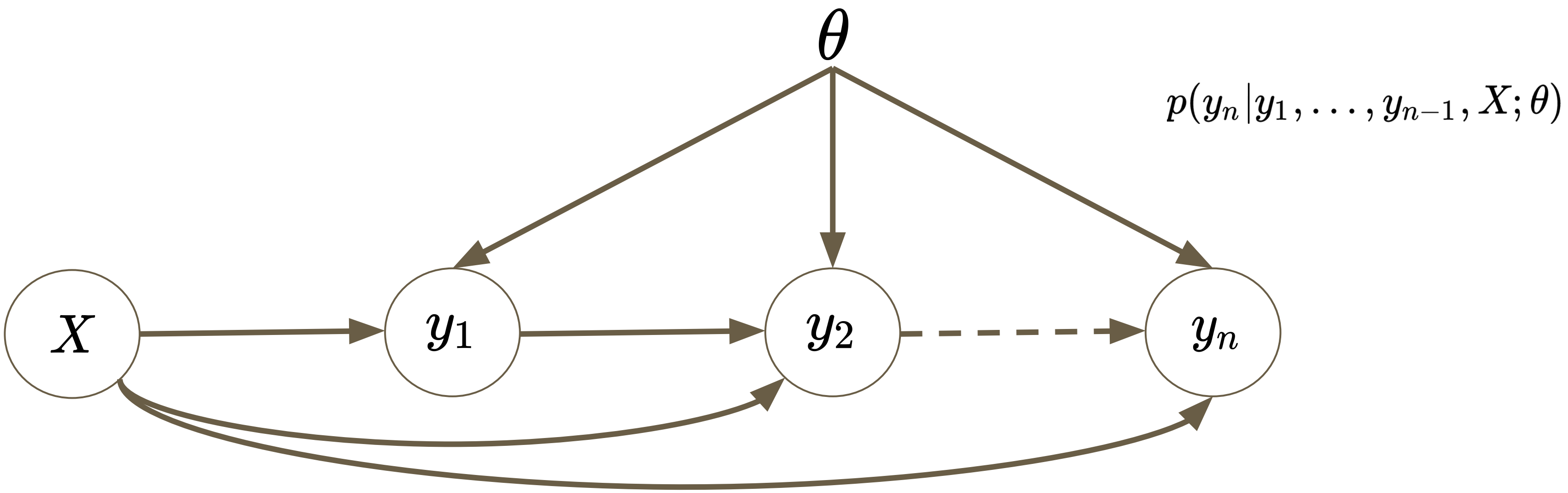
A conditional language model with no markov assumption



$$p(y_2 | y_1, X; \theta)$$

Summary: Generating Text using RNNs

A conditional language model with no markov assumption



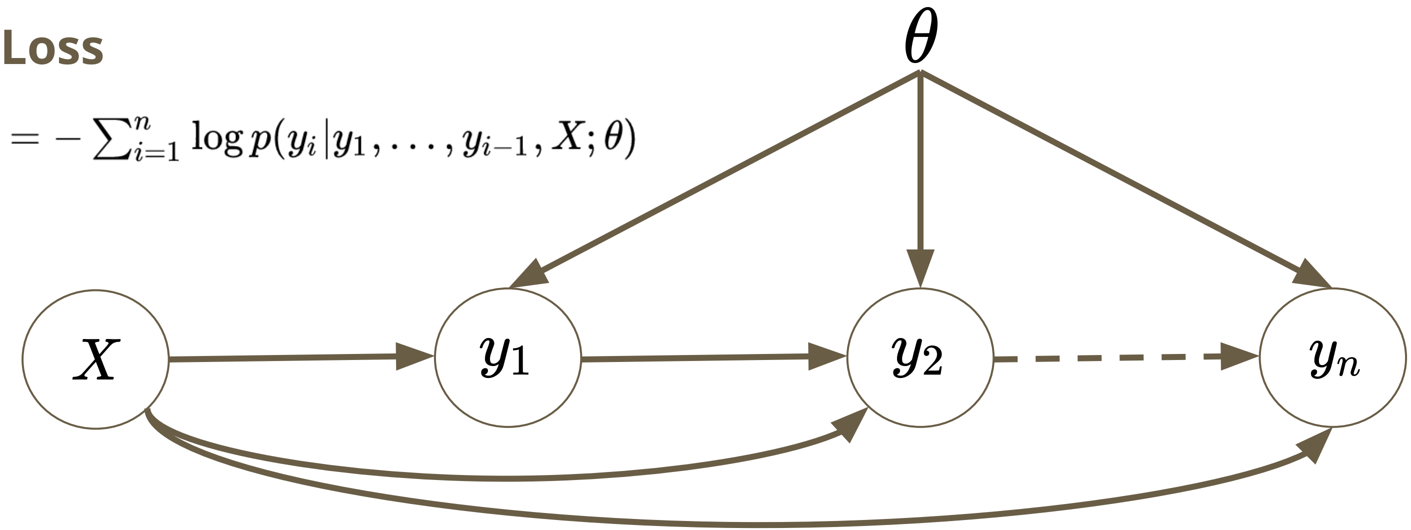
Summary: Generating Text using RNNs

Discriminative training: Learn to predict from a fixed vocabulary distribution

$$p(Y|X;\theta) = \prod_{i=1}^{n} p(y_i|y_1,\ldots,y_{i-1},X;\theta)$$

Cross Entropy Loss

$$L(\theta) = -\sum_{i=1}^{n} \log p(y_i|y_1,\ldots,y_{i-1},X;\theta)$$



Advantages over Pre-Neural Text Production Approaches

- End-to-end training
- Robustness and better generalization
- Leaving behind symbolic representations

Problems and Methods in Neural Text Production

- How to produce adequate texts ?
- How to model various inputs ?
- How to guide learning towards communication goals ?
- Datasets, Challenges and Open Issues

Decoding

Encoding

Producing Adequate Text

Some Problems with Neural Generation

ACCURACY: The output text sometimes contains information that does not match the input.

REPETITIONS: The output text sometimes contains repetitions

COVERAGE: The output text sometimes does not cover all the input

RARE OR UNKNOWN WORDS

DISFLUENT OUTPUT

Example from Recipe Generation (Kiddon et al, 2016)

Title: Almond-raspberry thumbprint cookies

Deleted

Ingredients: 1 cup butter or margarine softened, 1 cup sugar, 1 can solo almond filling, 2 egg yolks, 1 tsp almond extract, 2 1/2 cups all-purpose flour, 1/2 tsp baking powder, 1/2 tsp salt, 1 can solo raspberry filling

In a large bowl, beat butter and sugar until light and fluffy. Add eggs, one at a time, beating well after each addition. Blend in flour, baking powder, **baking soda**, salt and **baking soda**. Add to creamed mixture and mix well. Stir in nuts and vanilla. Drop dough by rounded table spoonfuls 2 " apart onto ungreased cookie sheets. Bake 8 to 10 minutes or until edges are lightly browned. Cool on wire racks.

Added **Repeated**

Example from Summarization

Lagos nigeria (CNN). a day after winning nigeria's presidency, muhammadu buhari told cnn's christiane amanpour that he plans to aggressively fight corruption that has long plagued nigeria and go after the root of the nation's unrest. buhari said he'll "rapidly give attention" to curbing violence in the northeast part of nigeria, where the terrorist group boko haram operates. by cooperating with neighboring nations chad, cameroon and niger, he said his administration is confident it will be able to thwart criminals and others contributing to nigeria's instability. for the first time in nigeria's history, the opposition defeated the ruling party in democratic elections. buhari defeated incumbent goodluck jonathan by about 2 million votes, according to nigeria's independent national electoral commission. the win comes after a long history of military rule, coups and botched attempts at democracy in africa's most populous nation.

RARE WORD

ADDED

Seq2Seq + Attention: UNK UNK says his administration is confident it will be able to destabilize nigeria's economy. UNK says his administration is confident it will be able to thwart criminals and other nigerians. he says the country has long nigeria and nigeria's economy.

Pointer-Gen: muhammadu buhari says he plans to aggressively fight corruption **in the northeast part of nigeria**. he says he'll "rapidly give attention" to curbing violence **in the northeast part of nigeria**. he says his administration is confident it will be able to thwart criminals.

See et al. 2017

Example from Generation

```
state
 :arg0 ( person
  :arg0-of ( have-org-role
   :arg1 ( committee :mod technical )
   :arg3 ( expert
    :arg1 person
    :arg2 missile
    :mod loc_0 ) ) )
 :arg1 ( evidence
  :arg0 equipment
  :arg1 ( plan :arg1 ( transfer :arg1 ( contrast
   :arg1 ( missile :mod ( just :polarity - ) )
    :arg2 ( capable
     :arg1 thing
     :arg2 ( make :arg1 missile ) ) ) ) )
  :mod ( impeach :polarity - :arg1 thing )
  :mod ( refute :polarity - :arg1 thing ) )
```

REF: A technical committee of indian missile experts stated that the equipement was unimpeachable and irrefutable evidence of a plan to transfer not just missiles but missile-making capabilities.

DELETED

SYS: A technical committee expert on the technical committee stated that the equipment is not impeached but it is not refutes. (Konstas et al. 2017)

DiSFLUENT

ADDED

Attention, Copy and Coverage

ATTENTION

- To **improve accuracy**

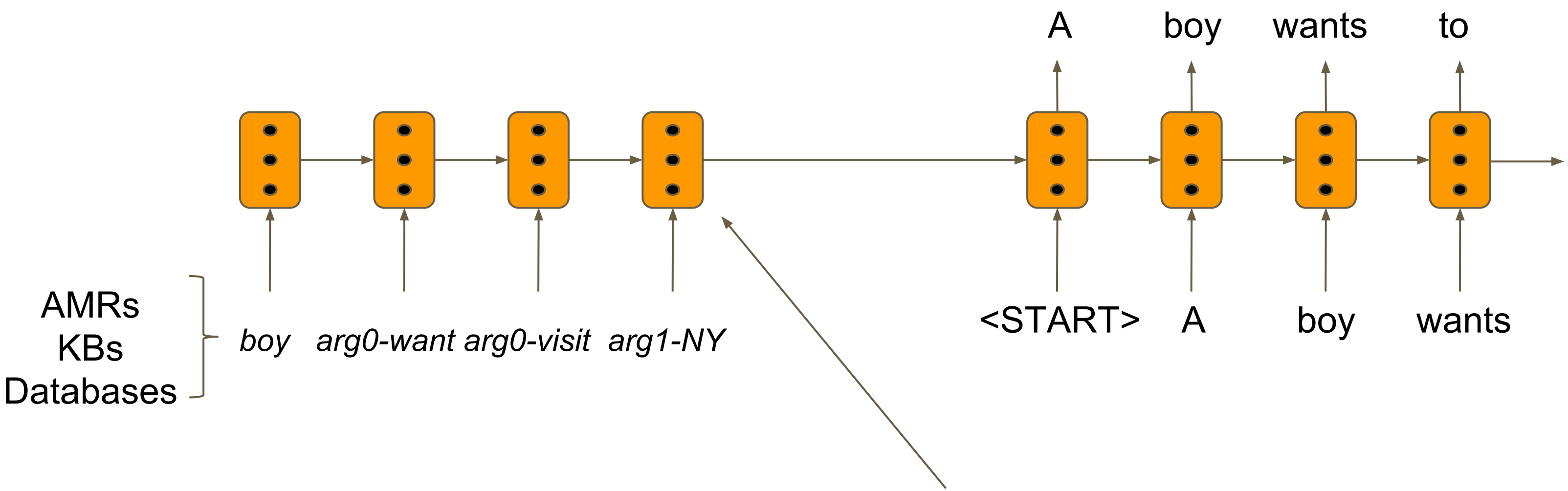
COPY

- To handle **rare or unknown words**
- To **copy** from the input

COVERAGE

- To help **cover all and only the input**
- To avoid **repetitions**

Encoder-Decoder without Attention



- The input is compressed into a fixed-length vector
- Performance decreases with the length of the input [Sutskever et al. 2014].

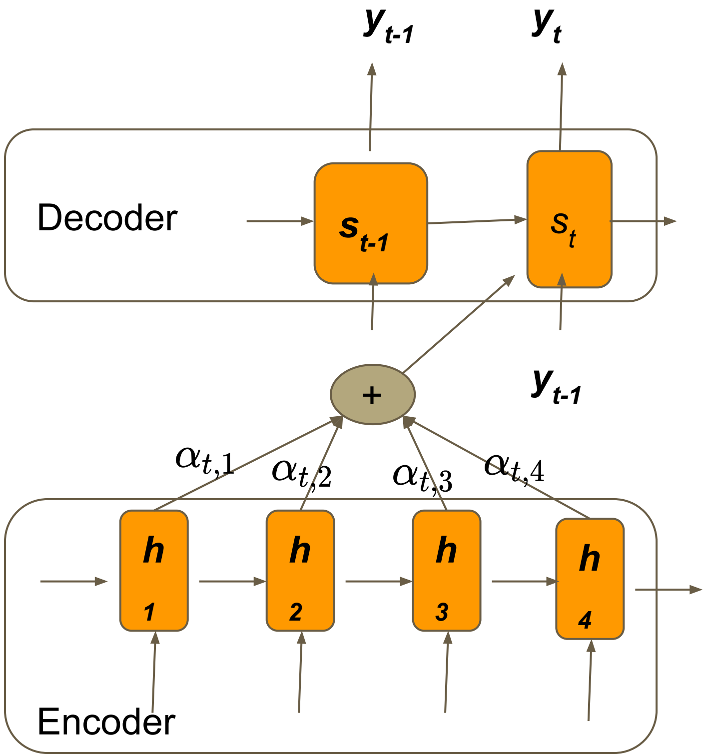
Encoder-Decoder with Attention

Takes as input the previous state s_{t-1} , the previously generated token y_{t-1} ***and a context vector*** c_t

This context vector

- depends on the previous state and therefore changes at each step
- Indicates which part of the input is most relevant to the decoding step

Encoder-Decoder with Attention



$$
\begin{aligned}
y_t &= softmax(W * s_t) \\
s_t &= f(s_{t-1}, y_{t-1}, c_t) \\
c_t &= \textstyle\sum_{j=1}^{T_x} \alpha_{t,j} h_j \\
\alpha_{t,j} &= \frac{exp(e_{t,j})}{\sum_{k=1}^{Tx} exp(e_{t,k})} \\
e_{t,j} &= a(s_{t-1}, h_j)
\end{aligned}
$$

The context vector $\boldsymbol{c_t}$ provides a representation of the input weighted by similarity with the current state.

It shows which part of the input is similar to the current decoding state

Badhanau et al. 2014

Copy

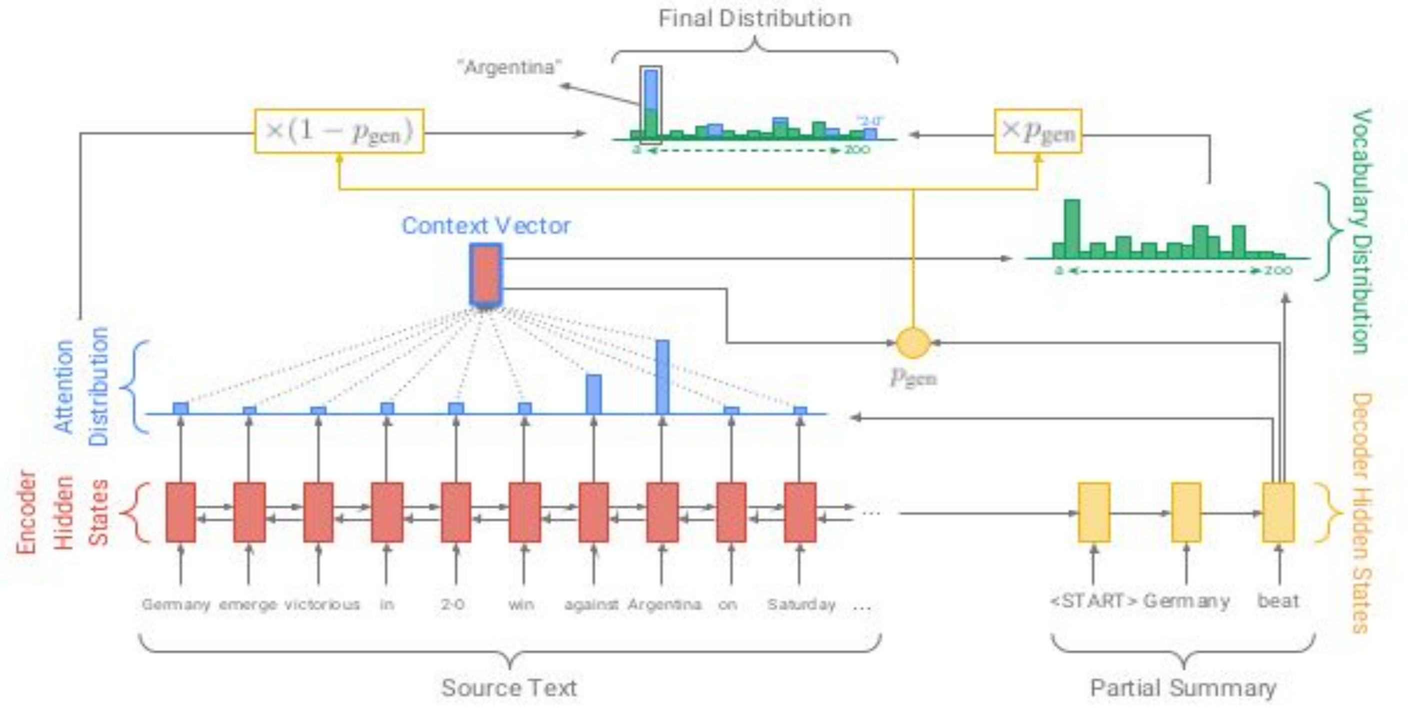
Motivation

- To copy from the input
- To handle rare or unknown words

Method

- Output words are taken either from the target vocabulary or from the input
- At each time step, the model decides whether to copy from the input or to generate from the target vocabulary

Modeling Document as a sequence of Tokens



See et al. 2017

Copying vs. Generating

Generation Probability

$$p_{gen} = \sigma(W_c * c_t + W_s * s_t + W_y * y_{t-1})$$

Probability of outputting word w

Attention distribution
0 if **w is not in Input**

$$p_{gen} \cdot P_{vocab}(w) + (1 - p_{gen}) \sum_{i:w=w_i} \alpha_{t,i}$$

Vocabulary distribution
0 if **w is not in VOCAB**

Copy and Generate

Lagos nigeria (CNN). a day after winning nigeria's presidency, muhammadu buhari told cnn's christiane amanpour that he plans to aggressively fight corruption that has long plagued nigeria and go after the root of the nation's unrest. buhari said he'll "rapidly give attention" to curbing violence in the northeast part of nigeria, where the terrorist group boko haram operates. by cooperating with neighboring nations chad, cameroon and niger, he said his administration is confident it will be able to thwart criminals and others contributing to nigeria's instability. for the first time in nigeria's history, the opposition defeated the ruling party in democratic elections. buhari defeated incumbent goodluck jonathan by about 2 million votes, according to nigeria's independent national electoral commission. the win comes after a long history of military rule, coups and botched attempts at democracy in africa's most populous nation.

Pointer-Gen: muhammadu buhari says he plans to aggressively fight corruption in the northeast part of nigeria. he says he'll "rapidly give attention" to curbing violence in the northeast part of nigeria. he says his administration is confident it will be able to thwart criminals.

See et al. 2017

Copy and generate in Text Production

Paraphrasing and Simplification: [Ciao et al. AAAI 2017].

Text Summarisation: [Gu, Lu, Li, Li, ACL 2016], [Gulcehre, Ahn, Nallapati, Zhou, Bengio, ACL 2016]

Extractive Summarisation: [Cheng and Lapata. ACL 2016].

Answer Generation: [He, Liu, Liu, Zhao ACL 2017]

Copy, Delexicalisation and Character-Based RNN

The COPY mechanism helps handling **rare or unknown words** (proper names, dates)

Alternative approaches for handling these are

- Delexicalisation
- Character-Based Encoders

Delexicalisation

Slot values occurring in training utterances are replaced with a placeholder token representing the slot

At generation time, these placeholders are then copied over from the input specification to form the final output

Wen et al. 2015, 2016

Delexicalisation

inform(restaurant name = Au Midi, neighborhood= midtown, cuisine = french)

Au Midi is in Midtown and serves French food.

inform(restaurant name = **restaurant name**, neighborhood= **neighborhood**, cuisine = **cuisine**)

restaurant name is in **neighborhood** and serves **cuisine** food.

Character-Based Encoding

Uses the open source tf-seq2seq framework to train a **char2char model** on the E2E NLG Challenge data.

No delexicalization, lowercasing or even tokenization

Input semantics = sequence of characters

Human evaluation shows that

- The output is **grammatically perfect**
- The model **does not generate non words**

Agarwal et al. 2017

Coverage

Problem: Neural models tend to omit or repeat information from the input

Solution

- Use coverage as extra input to attention mechanism
- Coverage: cumulative attention, what has been attended to so far
- Penalise attending to input that has already been covered

Tu et al. 2017

Coverage in Summarisation

A ***Coverage Vector*** captures how much attention each input words has received

$$k_{t,j} = \sum_{t'=0}^{t-1} \alpha_{t'}$$

The attention mechanism is modified to take coverage into account

$$e_{t,j} = a(s_t, h_j, k_{t,j})$$

The loss is modified to penalise any overlap between the coverage vector and the attention distribution

$$loss_t = -logP(w_t) + \lambda \sum_j min(\alpha_{t,j}, k_{t,j})$$

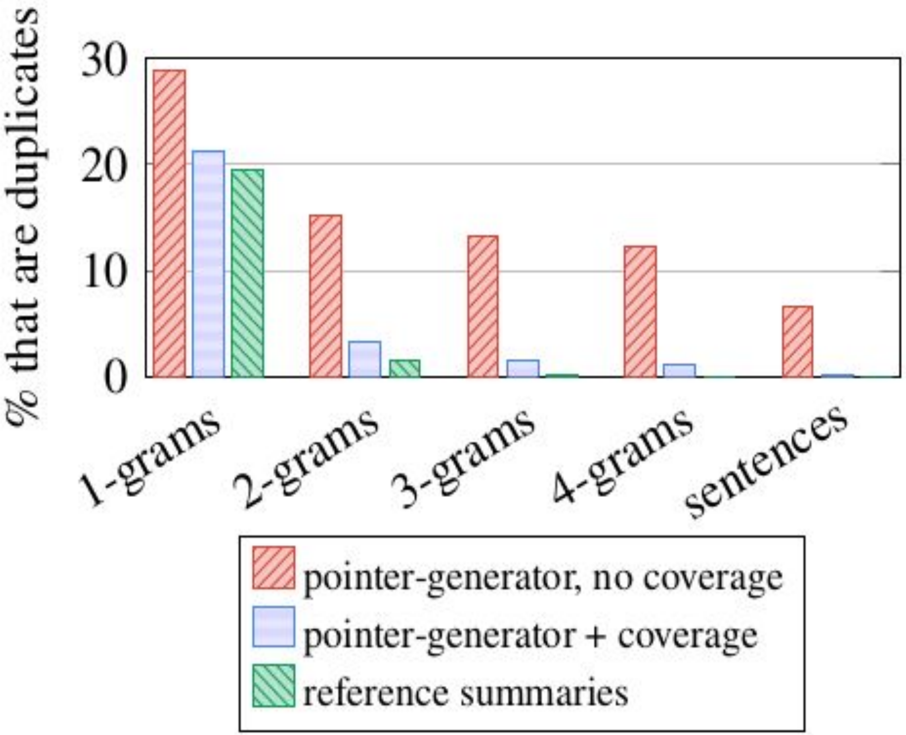
Summarising with coverage

Lagos nigeria (CNN). a day after winning nigeria's presidency, muhammadu buhari told cnn's christiane amanpour that he plans to aggressively fight corruption that has long plagued nigeria and go after the root of the nation's unrest. buhari said he'll "rapidly give attention" to curbing violence in the northeast part of nigeria, where the terrorist group boko haram operates. by cooperating with neighboring nations chad, cameroon and niger, he said his administration is confident it will be able to thwart criminals and others contributing to nigeria's instability. for the first time in nigeria's history, the opposition defeated the ruling party in democratic elections. buhari defeated incumbent goodluck jonathan by about 2 million votes, according to nigeria's independent national electoral commission. the win comes after a long history of military rule, coups and botched attempts at democracy in africa's most populous nation.

Pointer-Gen: muhammadu buhari says he plans to aggressively fight corruption in the northeast part of nigeria. he says he'll "rapidly give attention" to curbing violence in the northeast part of nigeria. he says his administration is confident it will be able to thwart criminals.

Pointer-Gen-Cov: muhammadu buhari says he plans to aggressively fight corruption that has long plagued nigeria. he says his administration is confident it will be able to thwart criminals. the win comes after a long history of military rule, coups and botched attempts at democracy in africa's most populous nation.

Summarising with coverage



Coverage successfully eliminates repetitions.

The proportion of duplicate n-grams is similar in the reference summaries and in the summaries produced by the model with coverage.

Coverage in Dialog: SC-LSTM

cells	i_t	$=$	$\sigma(W_{wi} w_t + W_{hi} h_{t-1})$
	f_t	$=$	$\sigma(W_{wf} w_t + W_{hf} h_{t-1})$
	o_t	$=$	$\sigma(W_{wo} w_t + W_{ho} h_{t-1})$
Reading Gate	r_t	$=$	$\sigma(W_{wr} w_t + W_{hr} h_{t-1})$
Candidate Cell	$\hat{c}_t$	$=$	$tanh(W_{wc} w_t + W_{hc} h_{t-1})$
DA Vector	d_t	$=$	$r_t \odot d_{t-1}$
New Cell	c_t	$=$	$i_t \odot \hat{c}_t + f_t \odot c_{t-1} + tanh(W_{dc} d_t)$
New hidden state	h_t	$=$	$o_t \odot tanh(c_t)$

Wen et al. 2015

Coverage in Dialog: SC-LSTM

cells $i_t = \sigma(W_{wi} w_t + W_{hi} h_{t-}$

$f_t = \sigma(W_{wf} w_t + W_{hf}$

$o_t = \sigma(W_{wo} w_t + W_{h}$

Reading Gate $r_t = \sigma(W_{wr} w_t + W_{hr}$

Conditioning Generation on the dynamically updated Dialog Move

Candidate Cell $\hat{c}_t = tanh(W_{wc} w_t + W_{hc}$

DA Vector $d_t = r_t \odot d_{t-1}$

New Cell $c_t = i_t \odot \hat{c}_t + f_t \odot c_{t-1} + tanh(W_{dc} d_t)$

New hidden state $h_t = o_t \odot tanh(c_t)$

Coverage in Dialog: SC-LSTM

cells $\quad i_t = \sigma(W_{wi} w_t + W_{hi} h_{t-1})$

$f_t = \sigma(W_{wf} w_t + W_{hf} h_{t-1})$

$o_t = \sigma(W_{wo} w_t + W_{ho} h_{t-1})$

Reading Gate $\quad r_t = \sigma($ ...

Updating the Dialog Move

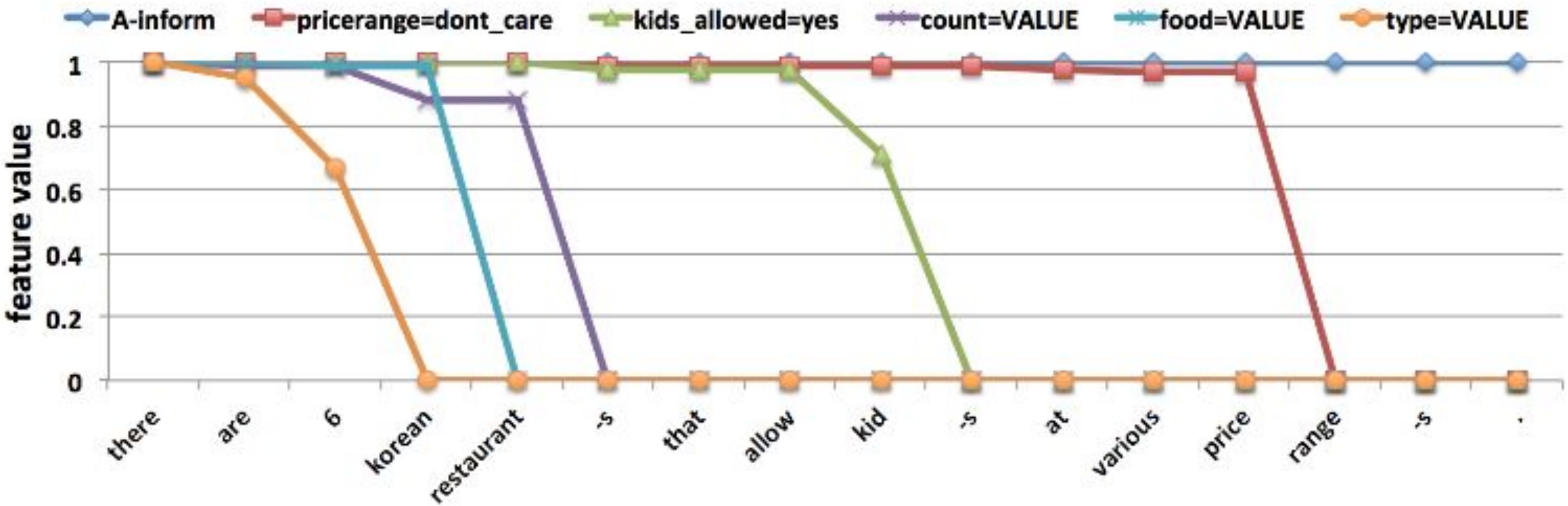
Candidate Cell $\quad \hat{c}_t = tanh($ ... $_{t-1})$

DA Vector $\quad d_t = r_t \odot d_{t-1}$

New Cell $\quad c_t = i_t \odot \hat{c}_t + f_t \odot c_{t-1} + tanh(W_{dc} d_t)$

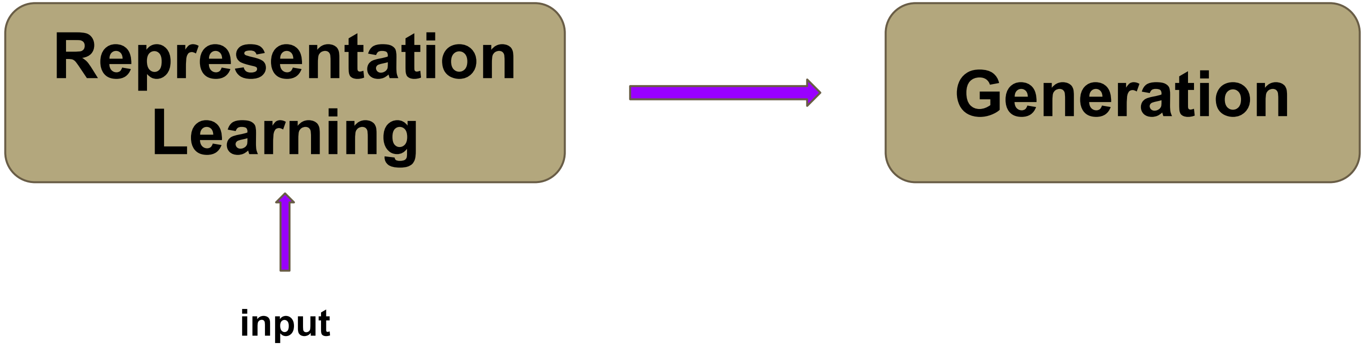
New hidden state $\quad h_t = o_t \odot tanh(c_t)$

Reading Gate in Action

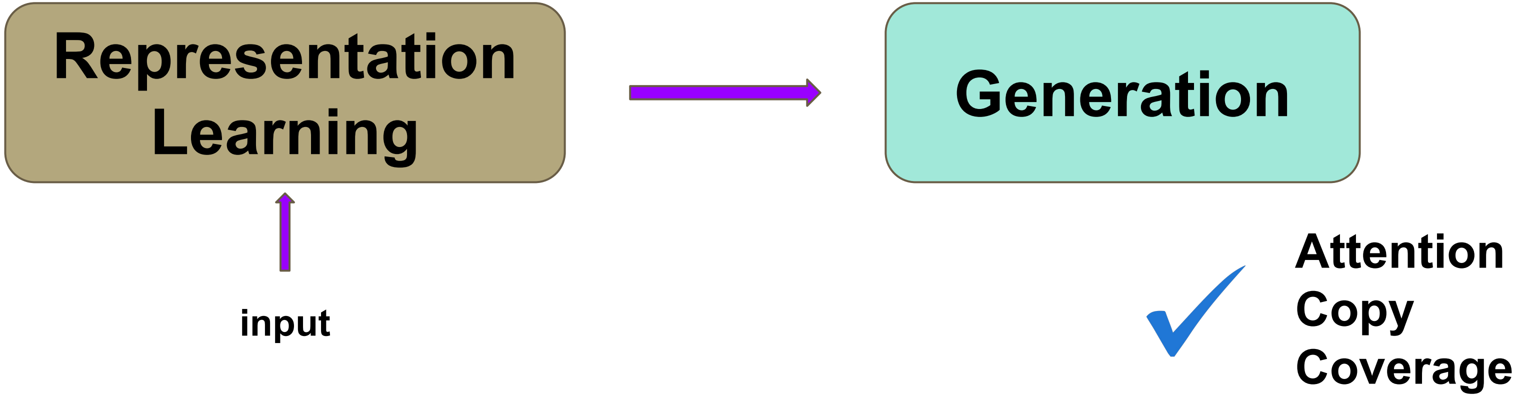


Text Production with Better Input Understanding

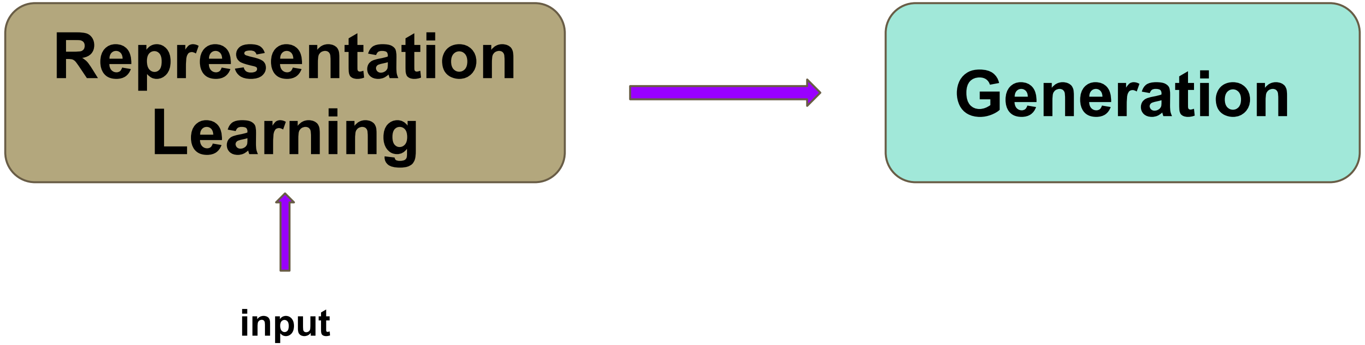
Deep Learning: A Uniform Framework for Text Production



Deep Learning: A Uniform Framework for Text Production



Deep Learning: A Uniform Framework for Text Production



Learning representations better suited for Input and Communication Goal

Taking Structure into account

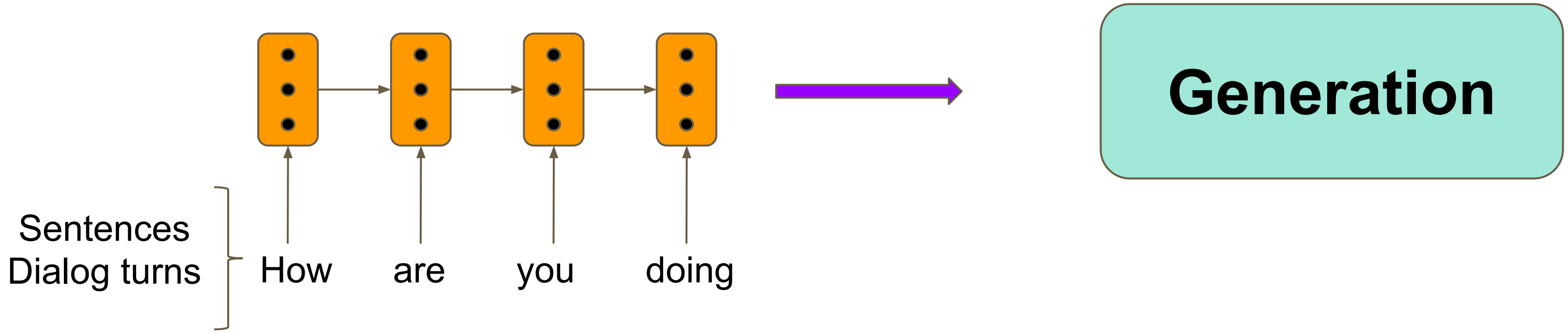
Text structure: Abstractive and extractive summarisation

- Hierarchical encoders
- Ensemble encoders
- Convolutional sentence encoders

Data structure: MR- and data-to-text Generation

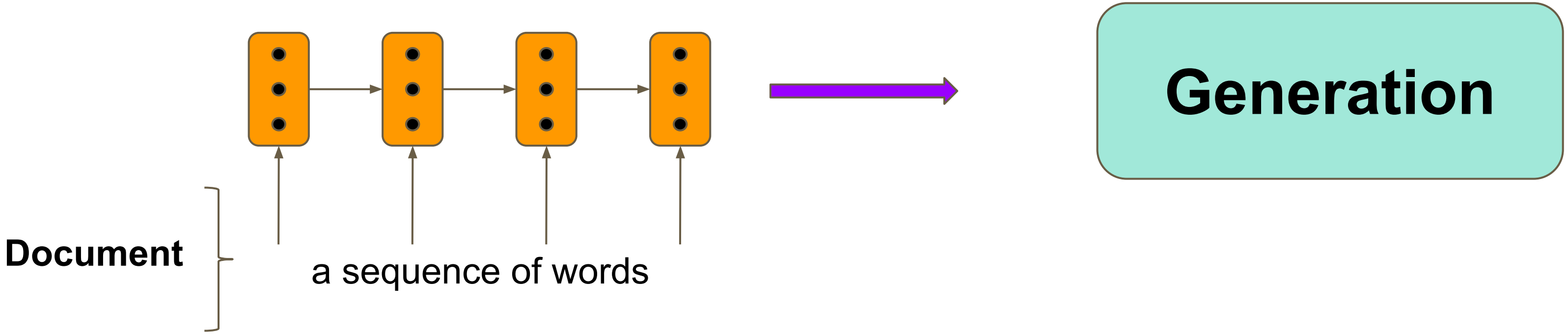
- Graph to sequence (AMR to text)
- Graph-Based Triple Encoder (RDF to text)
- Graph Convolutional Networks

Modeling Sentence as a sequence of Tokens



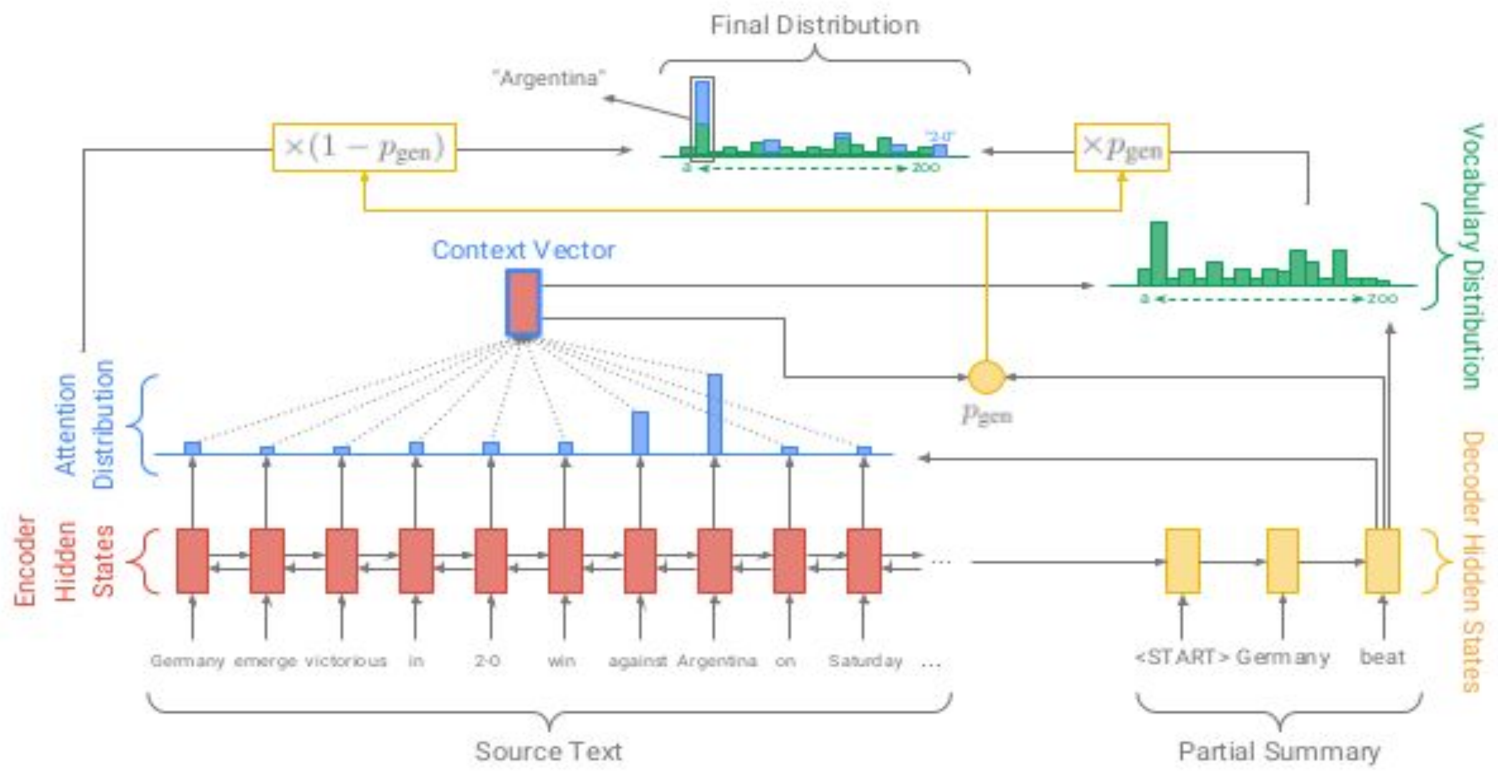
- **Sentence Simplification** (Zhang and Lapata, 2017)
- **Paraphrasing** (Mallinson at al. 2017)
- **Sentence Compression** (Filippova et al. 2015)
- **Conversation Model** (Li et al., 2016)

Modeling Document as a sequence of Tokens

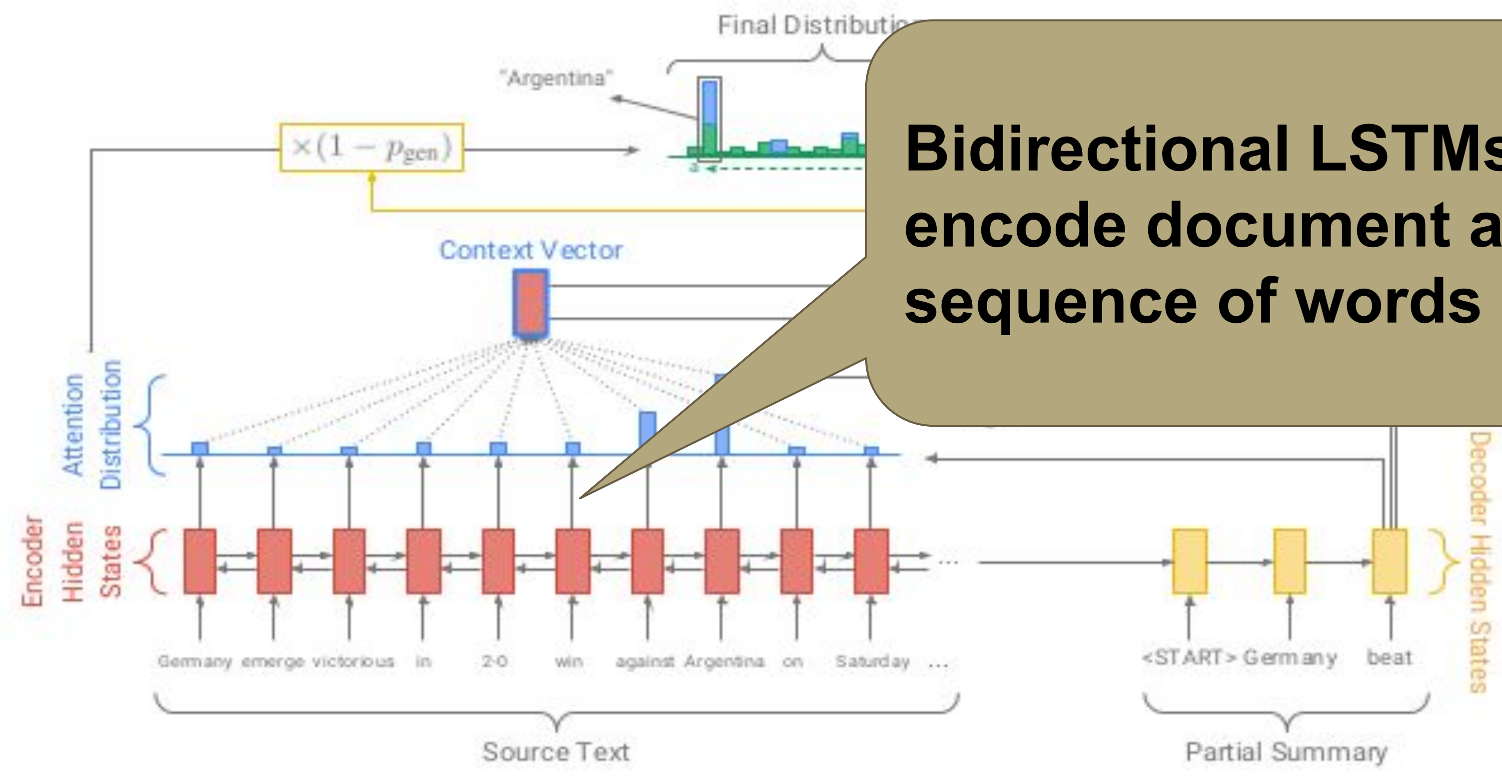


- **Abstractive Document Summarization** (Nallapati et al. 2016, See et al. 2017, Paulus et al. 2017, Pasunuru and Bansal 2018)

Modeling Document as a sequence of Tokens



Modeling Document as a sequence of Tokens



Bidirectional LSTMs to encode document as sequence of words

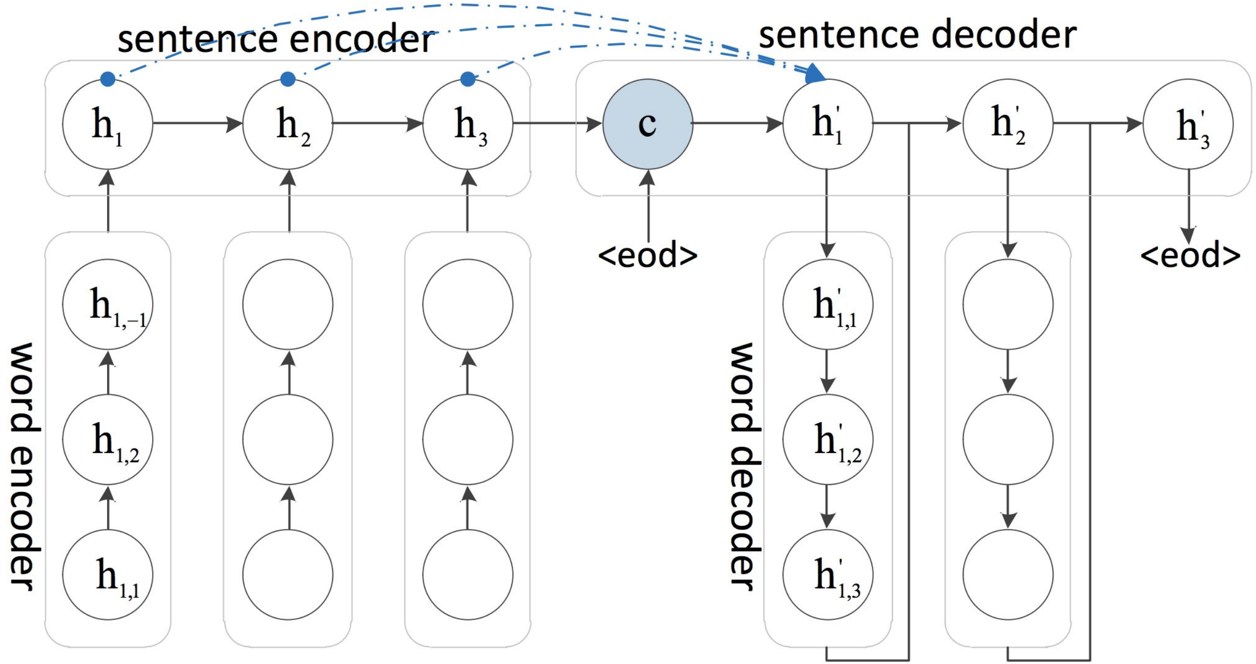
Modeling Document as a sequence of Tokens

- ✓ Simple sequential encoder
- ✓ Sequential Generators with copy, coverage and attention
- ✗ Ignores the hierarchical structure of a document
- ✗ Issues with long range dependencies

Hierarchical Document Encoders

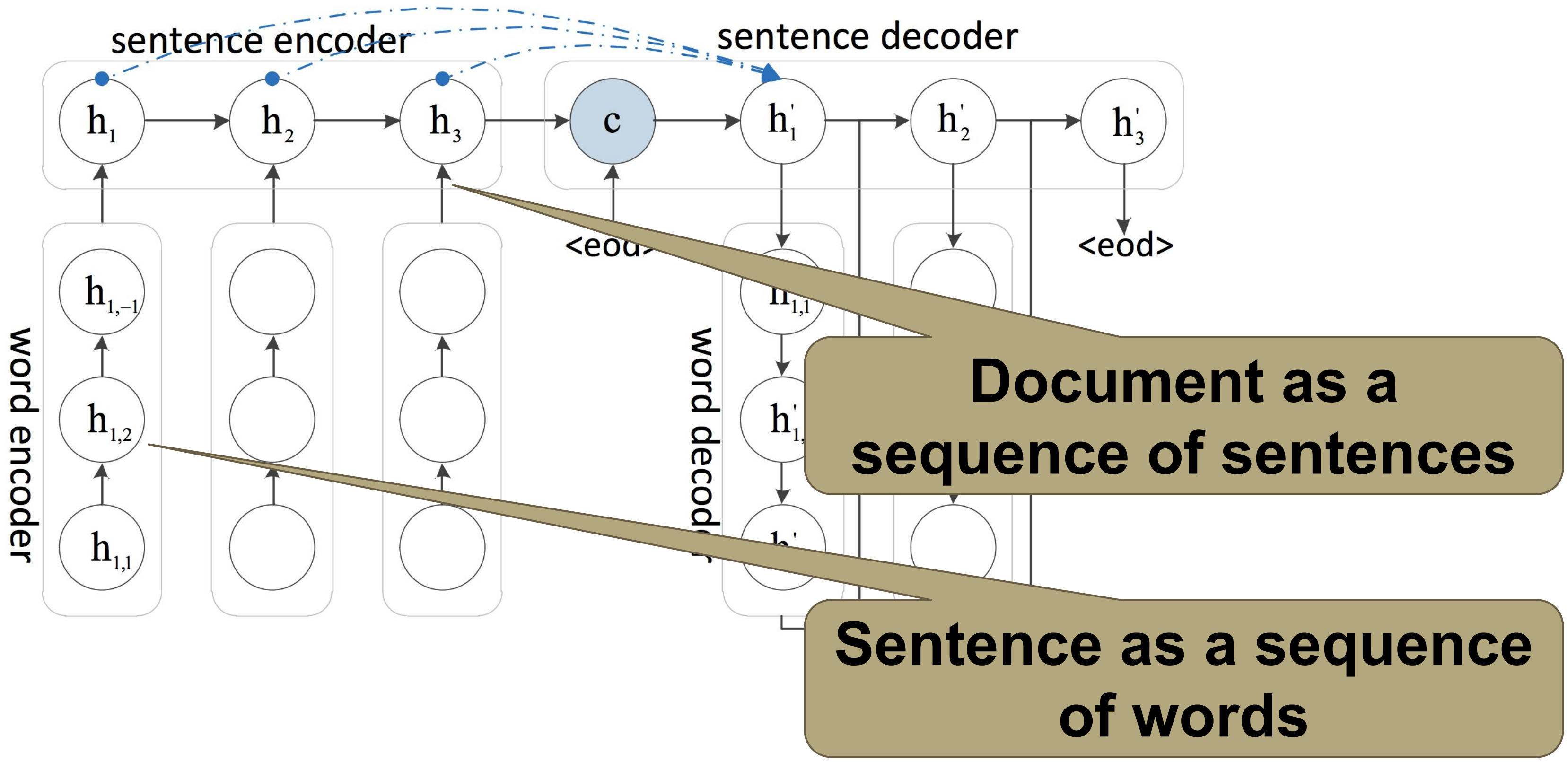
- ✓ Modeling document with sentence encoders (Cheng and Lapata, 2016, Tan et al. 2017, Narayan et al. 2018)
- ✓ Modeling document with paragraph encoders (Celikyilmaz et al. 2018)

Modeling Documents with Hierarchical LSTMs

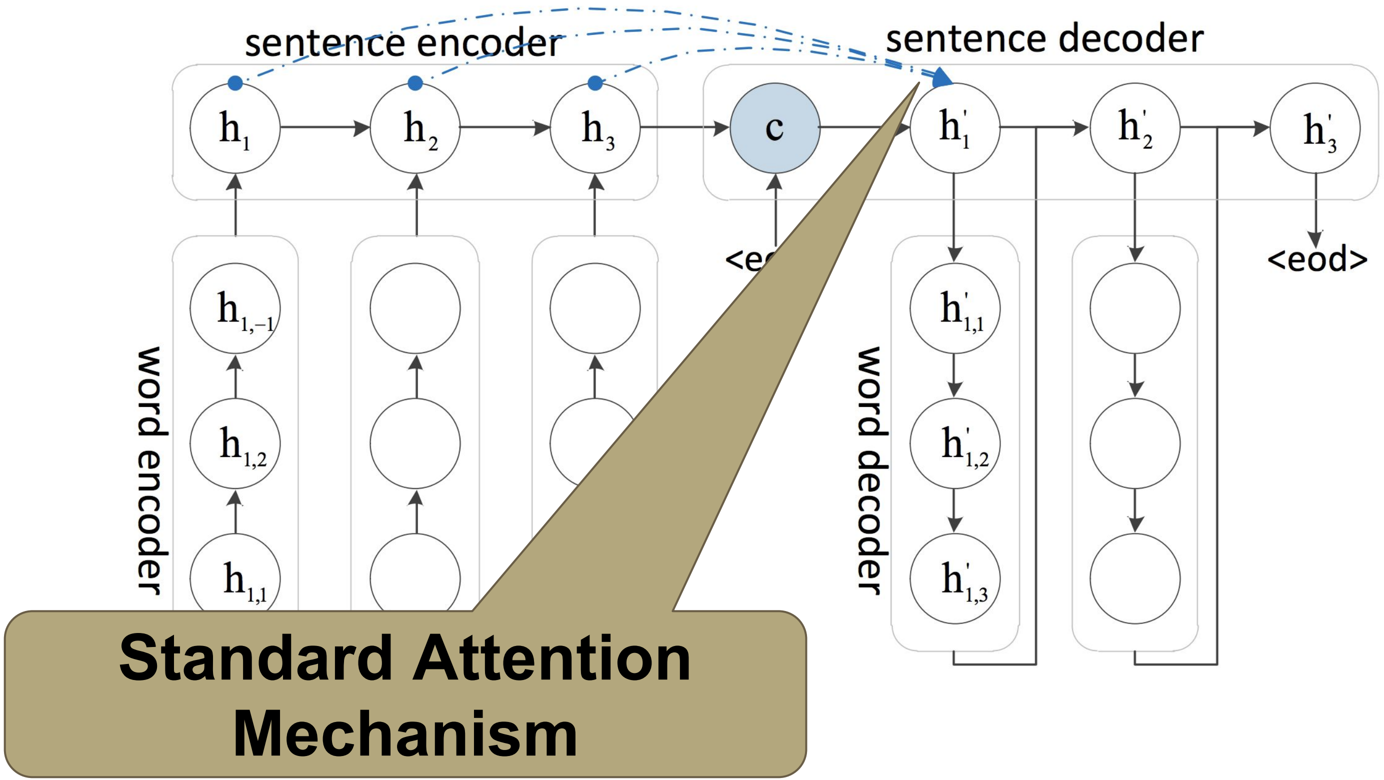


Abstractive Document Summarization (Tan et al. ACL 2017)

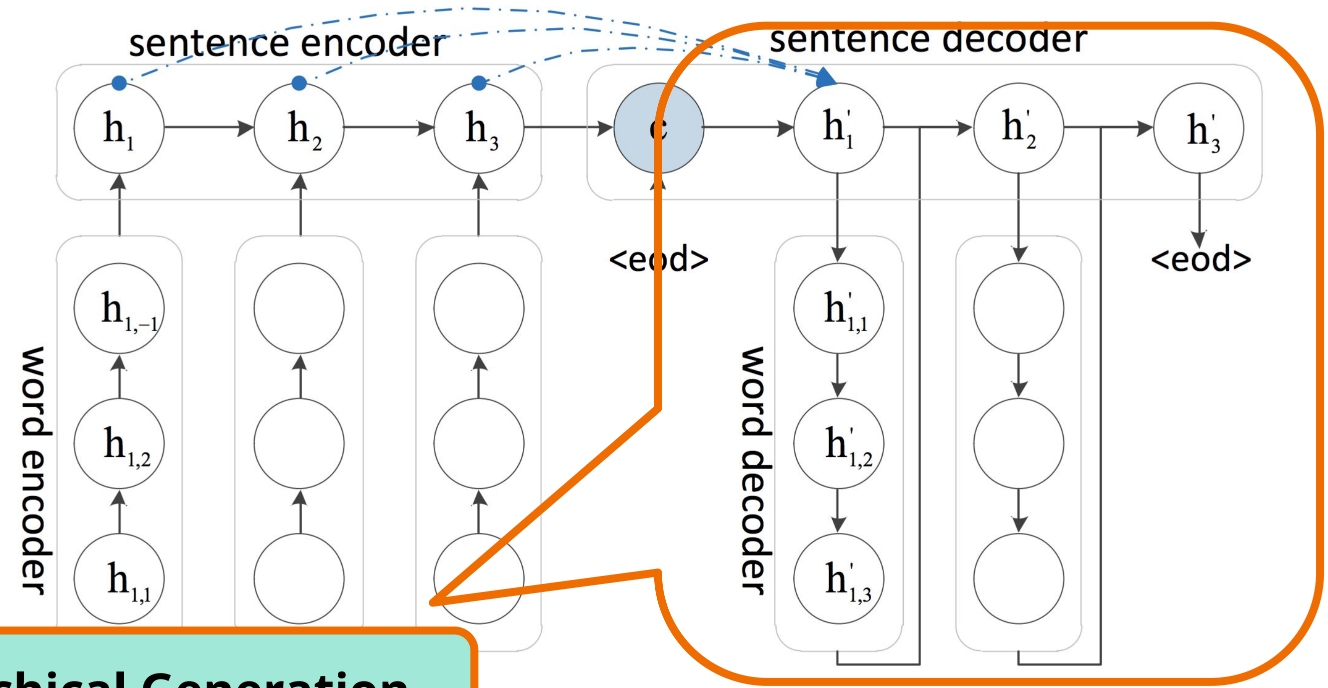
Modeling Documents with Hierarchical RNNs



Modeling Documents with Hierarchical LSTMs



Modeling Documents with Hierarchical RNNs



Hierarchical Generation

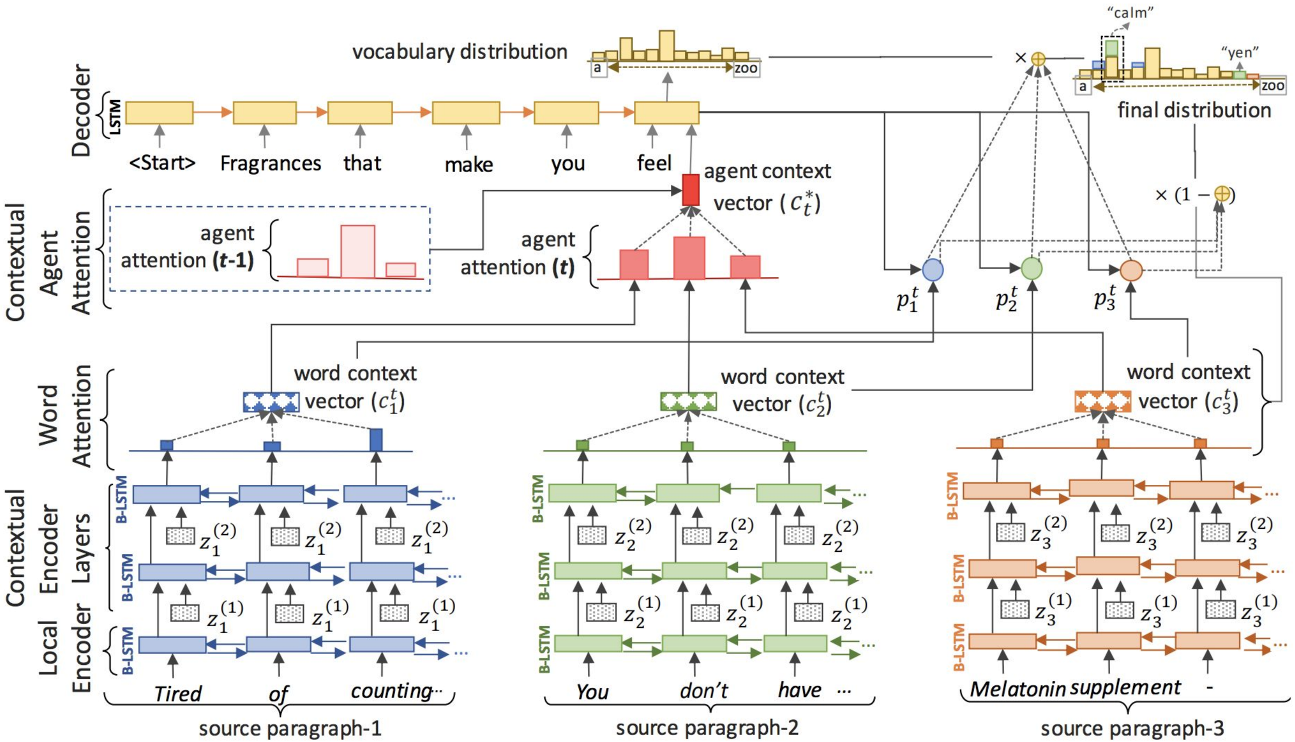
Modeling Documents with Hierarchical RNNs

Without a pointer generator mechanism, model suffers from generating out-of-vocabulary words.

It performs inferior to (See at al, ACL 2017).

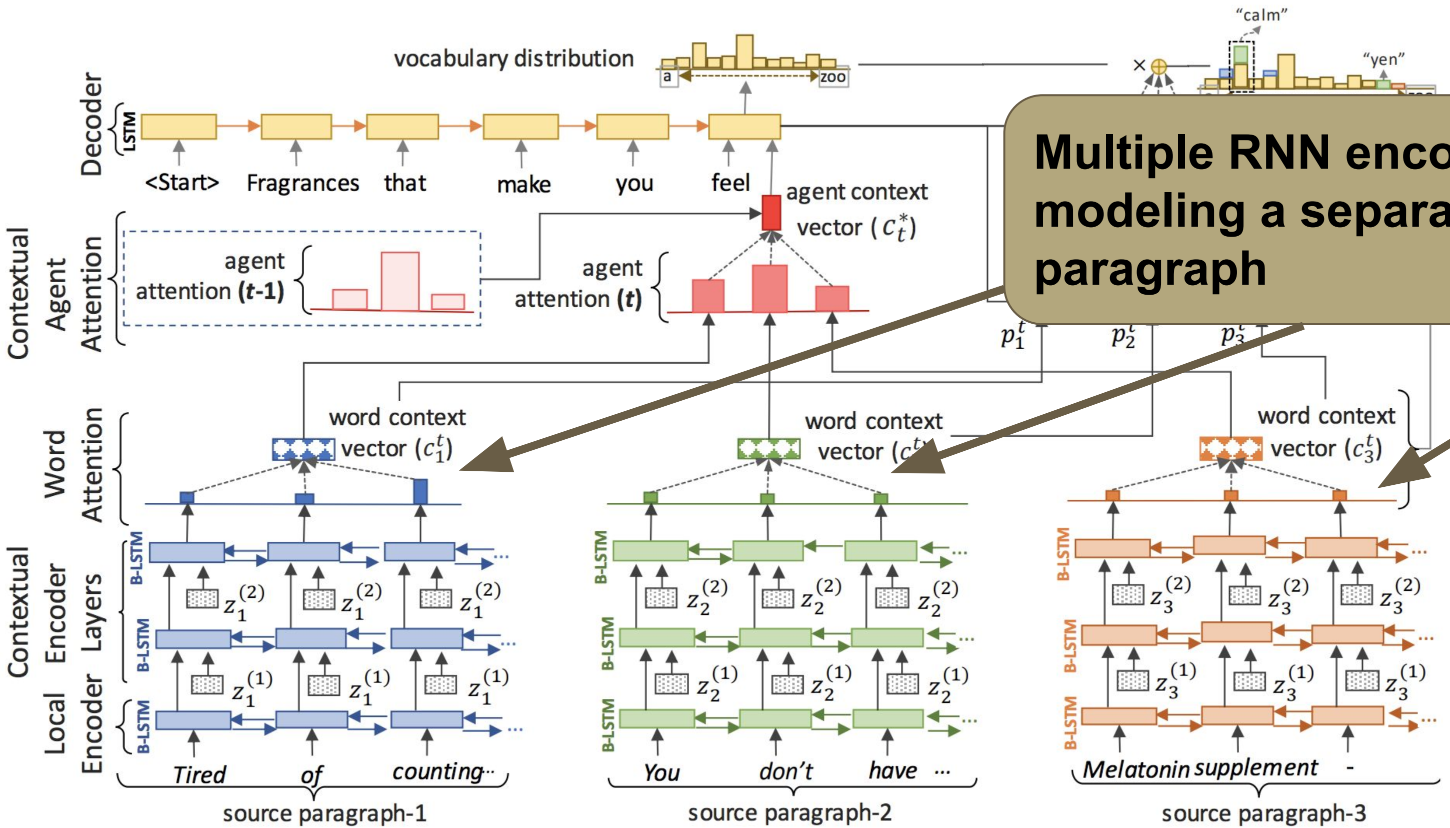
Abstractive Document Summarization (Tan et al. ACL 2017)

Modeling Document with Ensemble Encoders



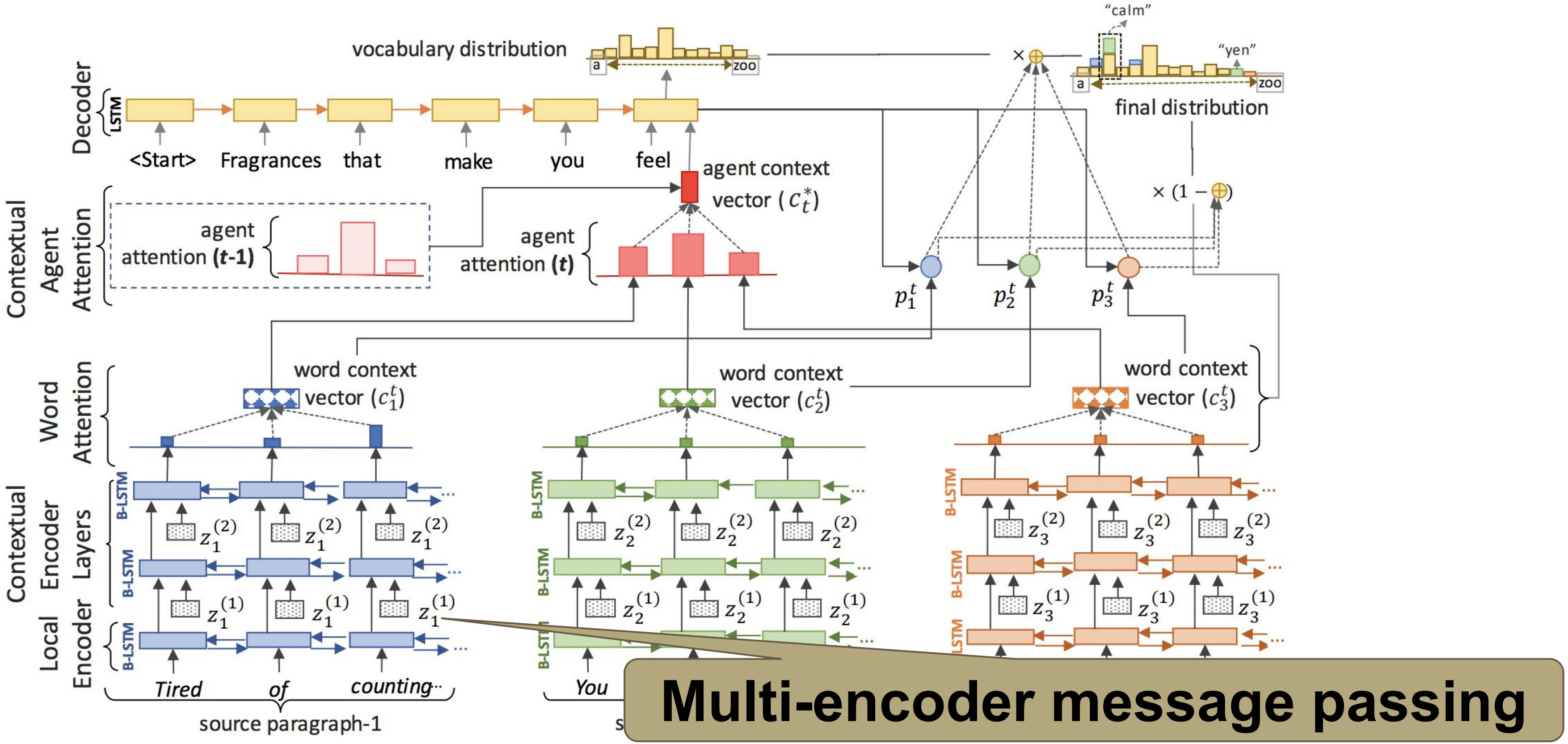
Abstractive Document Summarization (Celikyilmaz et al. NAACL 2018)

Modeling Document with Ensemble Encoders

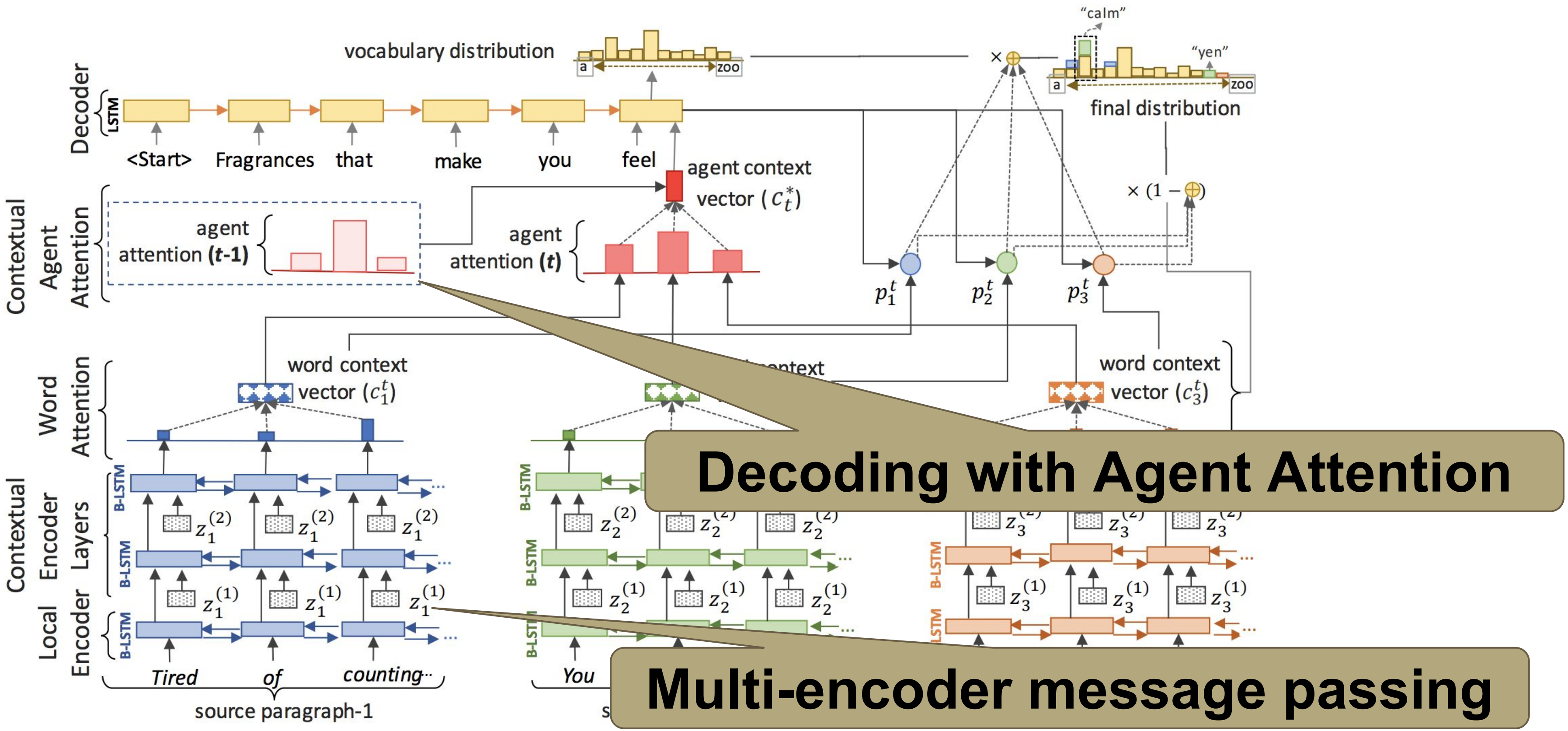


Multiple RNN encoders, each modeling a separate paragraph

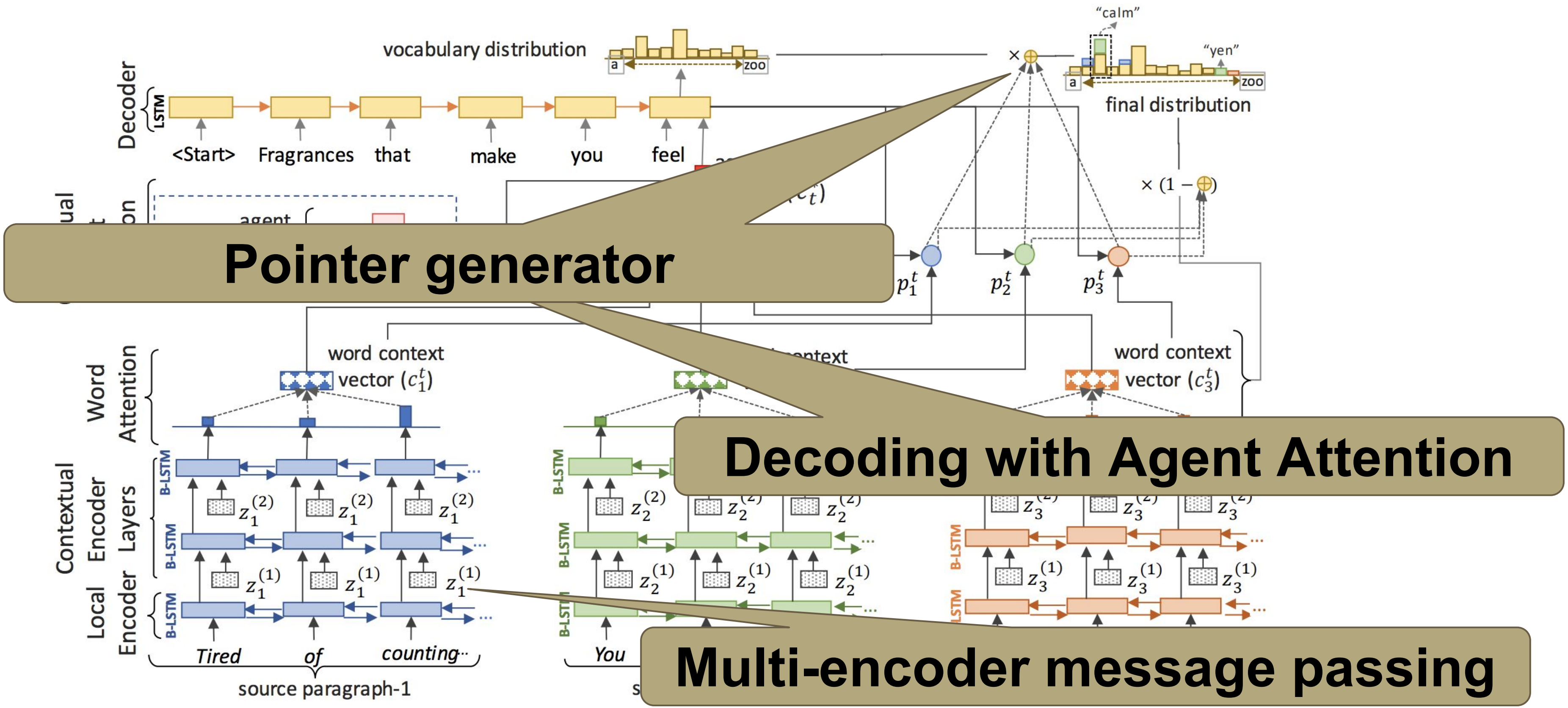
Modeling Document with Ensemble Encoders



Modeling Document with Ensemble Encoders



Modeling Document with Ensemble Encoders

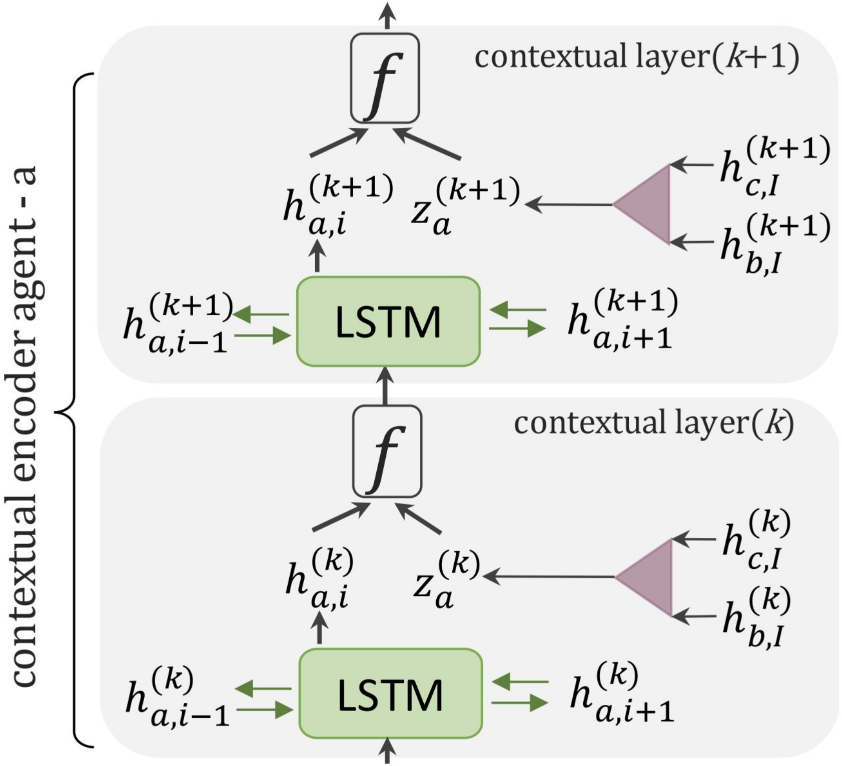


Multi-Encoder Message Passing

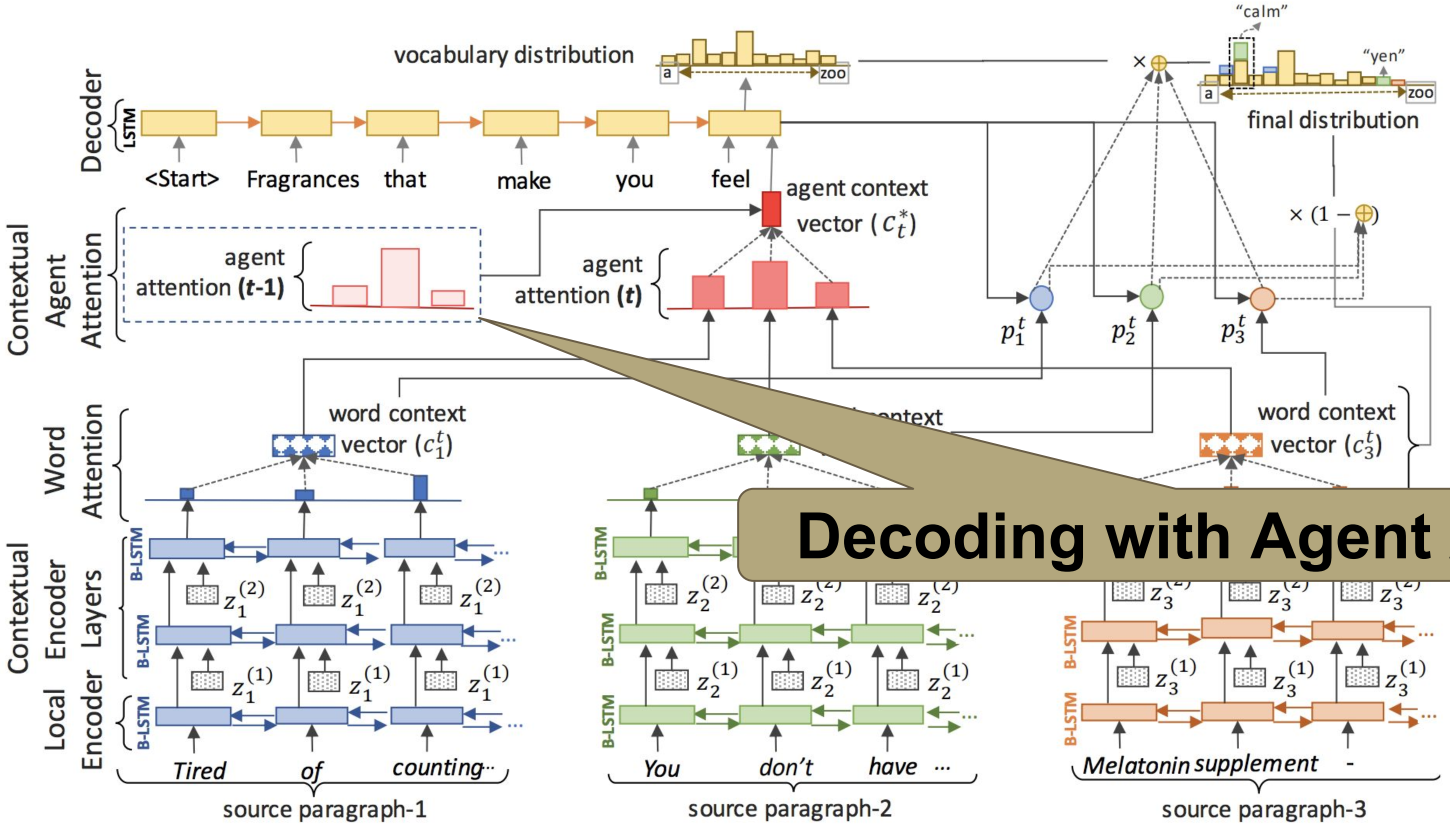
$$\overrightarrow{h}_i^{(k+1)}, \overleftarrow{h}_i^{(k+1)} = \mathrm{bLSTM}(f(h_i^{(k)}, z^{(k)}), \overrightarrow{h}_{i-1}^{(k+1)}, \overleftarrow{h}_{i+1}^{(k+1)})$$

$$h_i^{(k+1)} = W_2[\overrightarrow{h}_i^{(k+1)}, \overleftarrow{h}_i^{(k+1)}]$$

$$z^{(k)} = \frac{1}{M-1}\sum_{m \neq a} h_{m,I}^{(k)}$$



Modeling Document with Ensemble Encoders



Decoding with Agent Attention

Decoding with Hierarchical Attention

Word attention distribution for each paragraph

$$l_a^t = \text{softmax}(v_2^\intercal \tanh(W_5 h_a^{(K)} + W_6 s_t + b_1))$$

Decoder context

$$c_a^t = \sum_i l_{a,i}^t h_{a,i}^{(K)}$$

Document global agent attention distribution

$$g^t = \text{softmax}(v_3^\intercal \tanh(W_7 c^t + W_8 s_t + b_2))$$

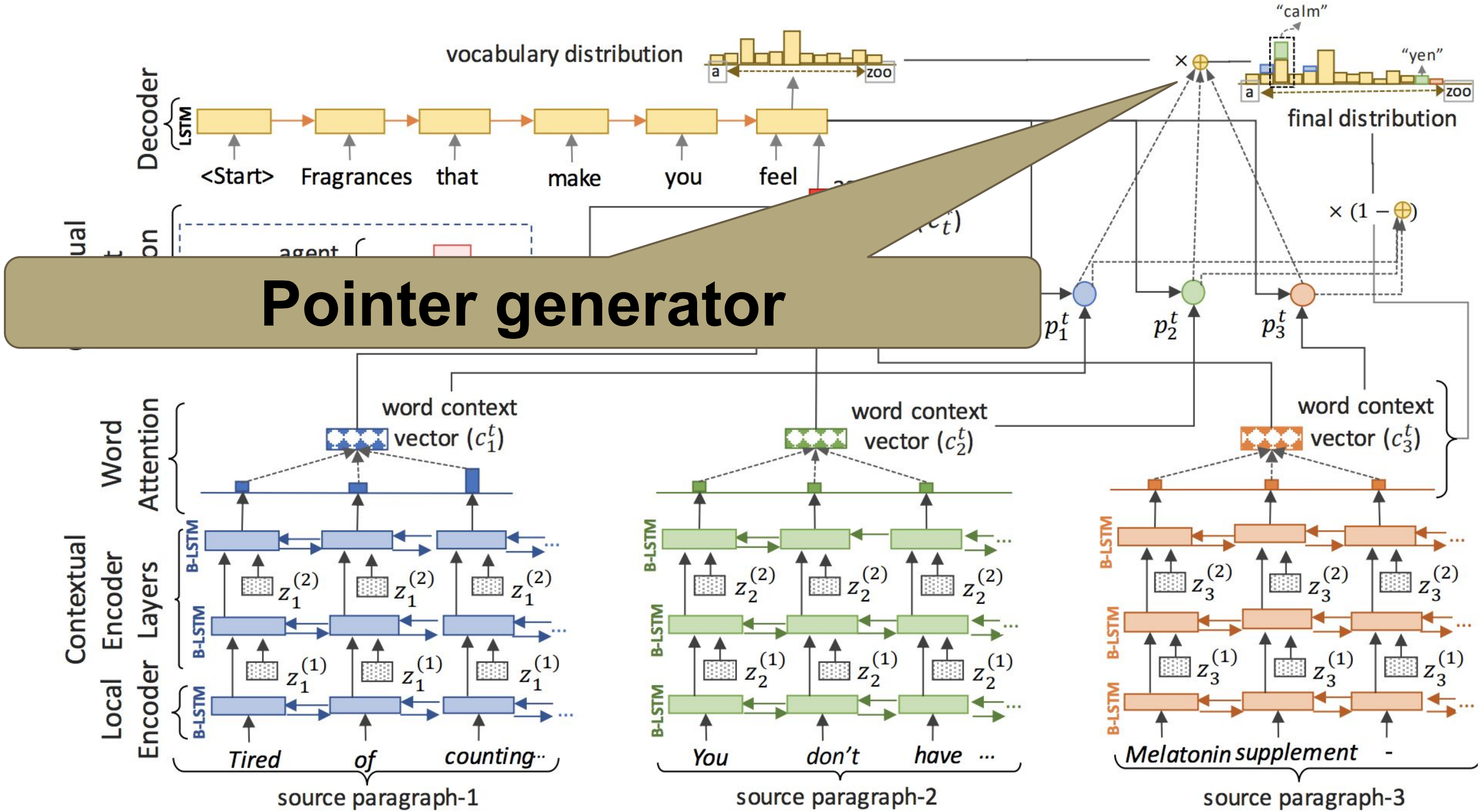
Agent context vector

$$c_t^* = \sum_a g_a^t c_a^t$$

Final vocabulary distribution

$$P^{voc}(w_t | s_t, w_{t-1}) = \text{softmax}(\text{MLP}([s_t, c_t^*]))$$

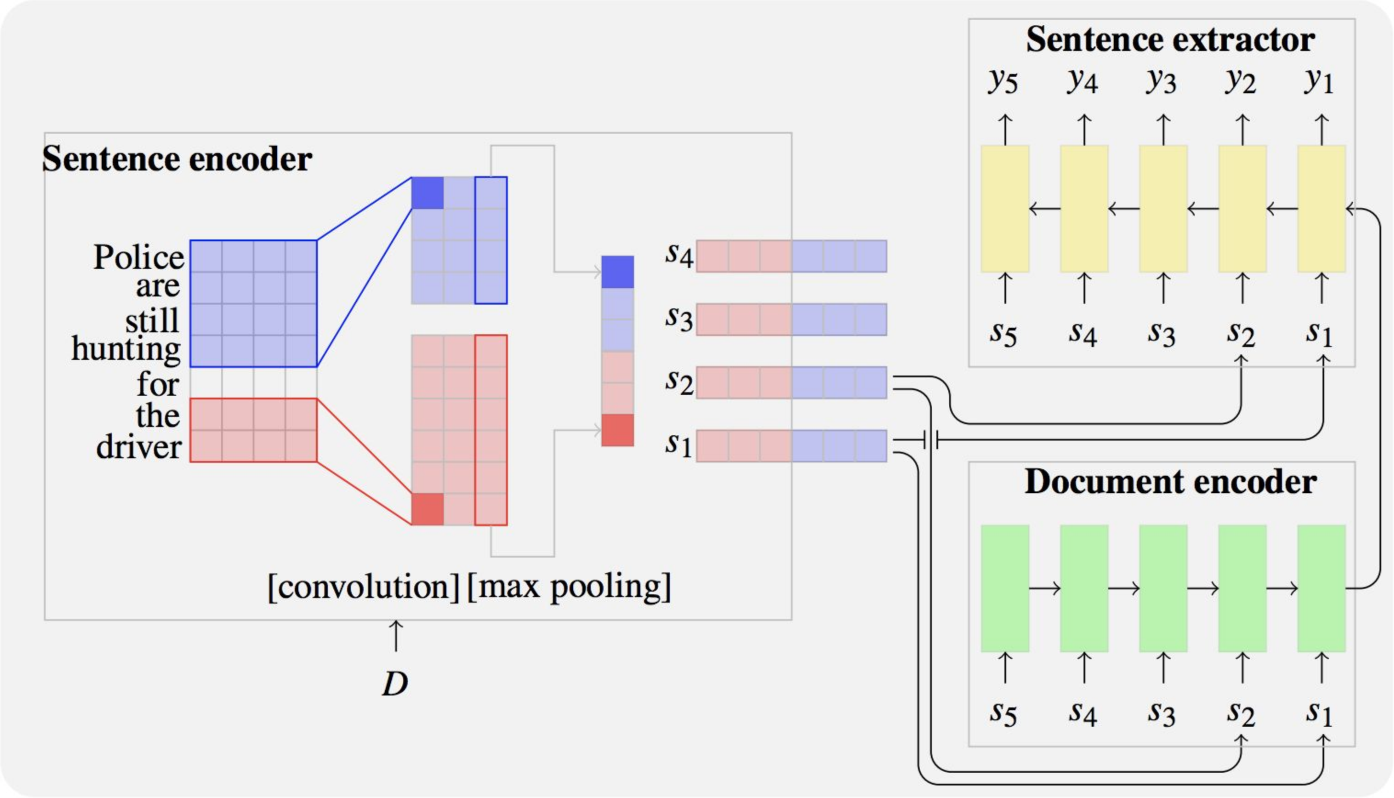
Modeling Document with Ensemble Encoders



Modeling Document with Ensemble Encoders

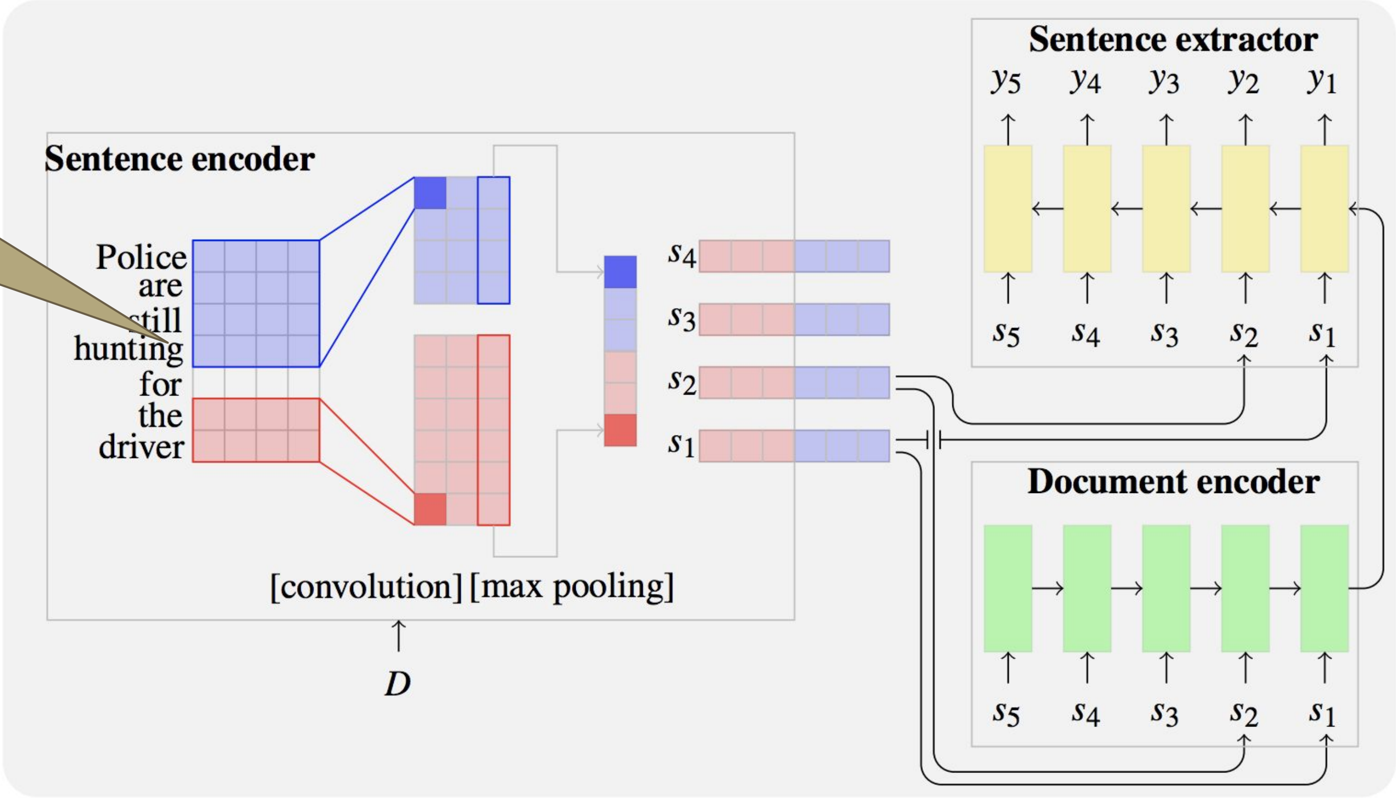
✓ Model achieves state-of-the-art performance outperforming (See at al, ACL 2017, Tan et al, ACL 2017).

Modeling Document With Convolutional Sentence Encoders

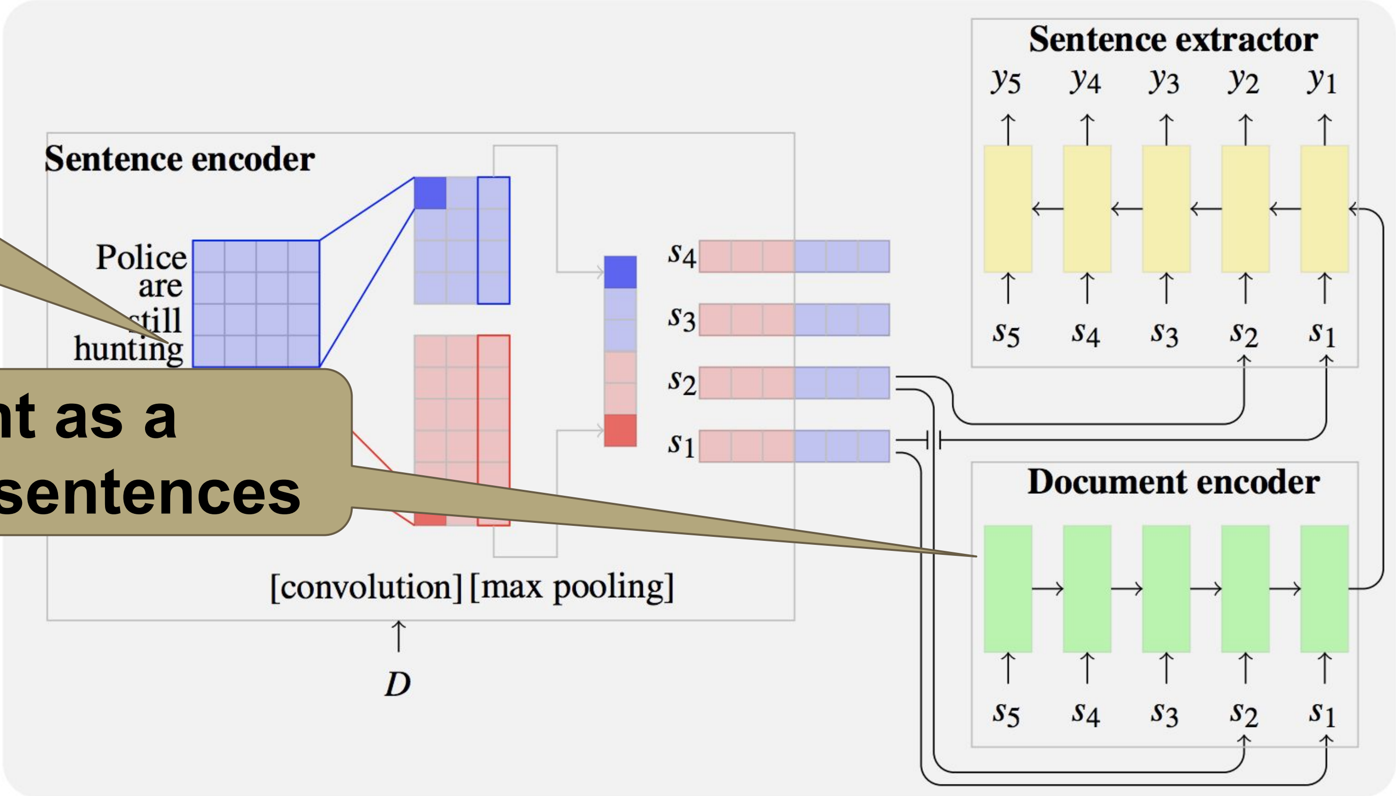


Modeling Document With Convolutional Sentence Encoders

Convolutional Sentence Encoder



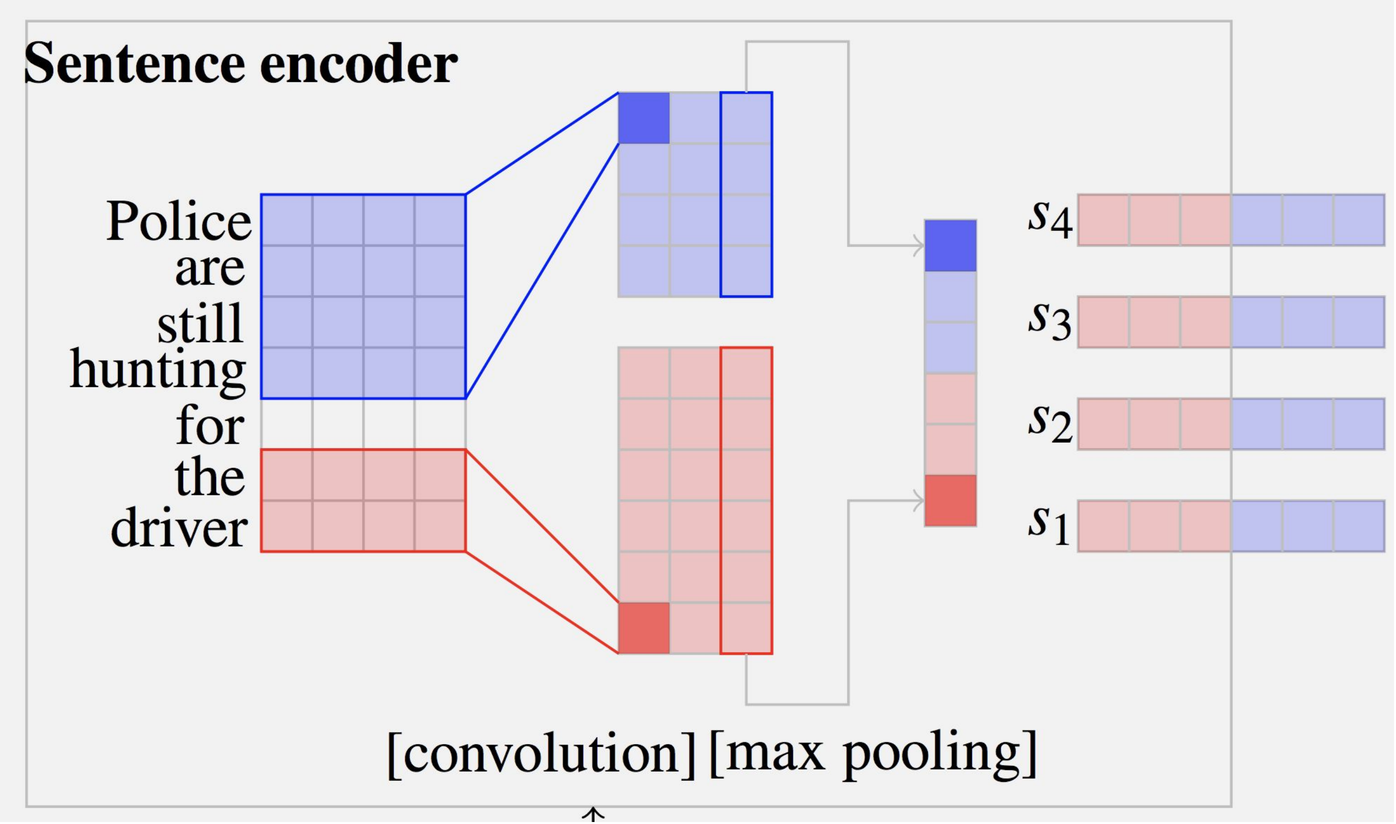
Modeling Document With Convolutional Sentence Encoders



Convolutional Sentence Encoder

Document as a sequence of sentences

Convolutional Sentence Encoders



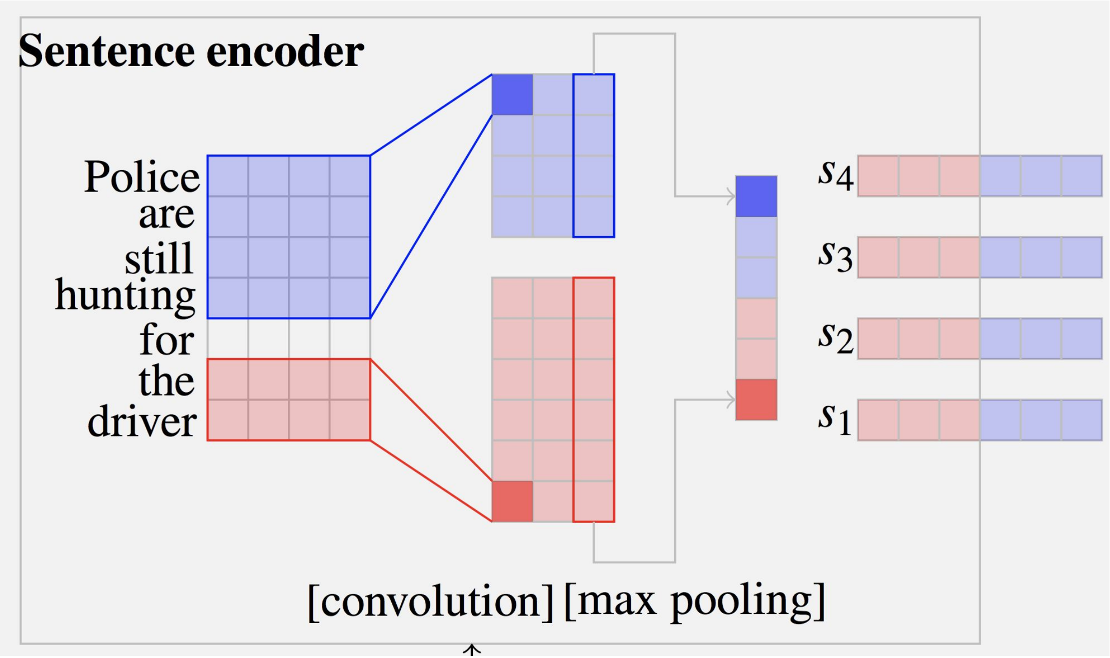
(Kim, EMNLP 2014)

Convolutional Sentence Encoders

Convolution: $$C = \text{conv1d}(S_n, W_d) \in \mathcal{R}^{n-d+1}$$

Non-linearity: $$Y = \sigma(C)$$

Max-pooling: $$y_{max} = \text{max-pool}(Y) \in R^1$$



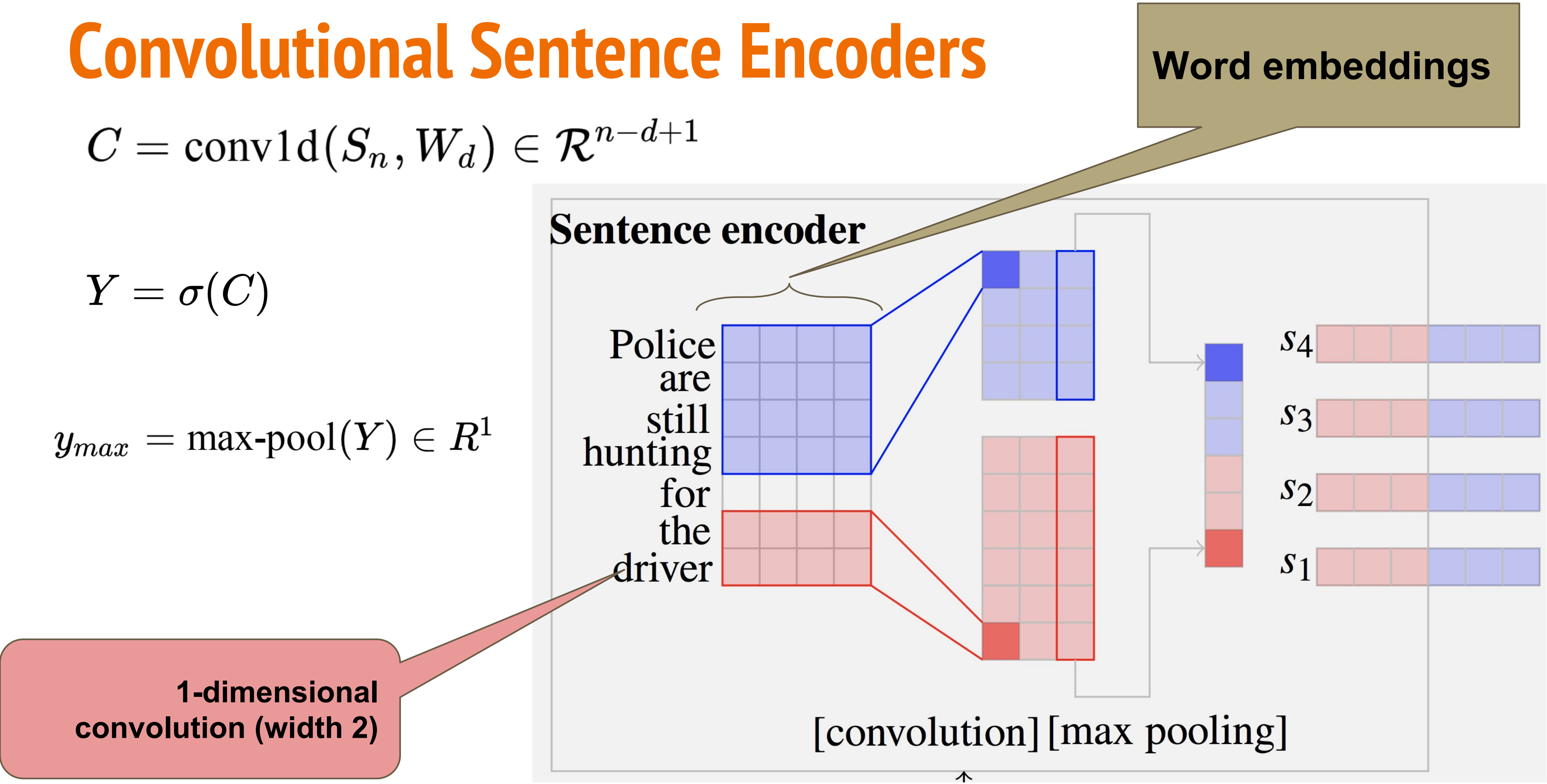
(Kim, EMNLP 2014)

Convolutional Sentence Encoders

$$C = \text{conv1d}(S_n, W_d) \in \mathcal{R}^{n-d+1}$$

$$Y = \sigma(C)$$

$$y_{max} = \text{max-pool}(Y) \in R^1$$

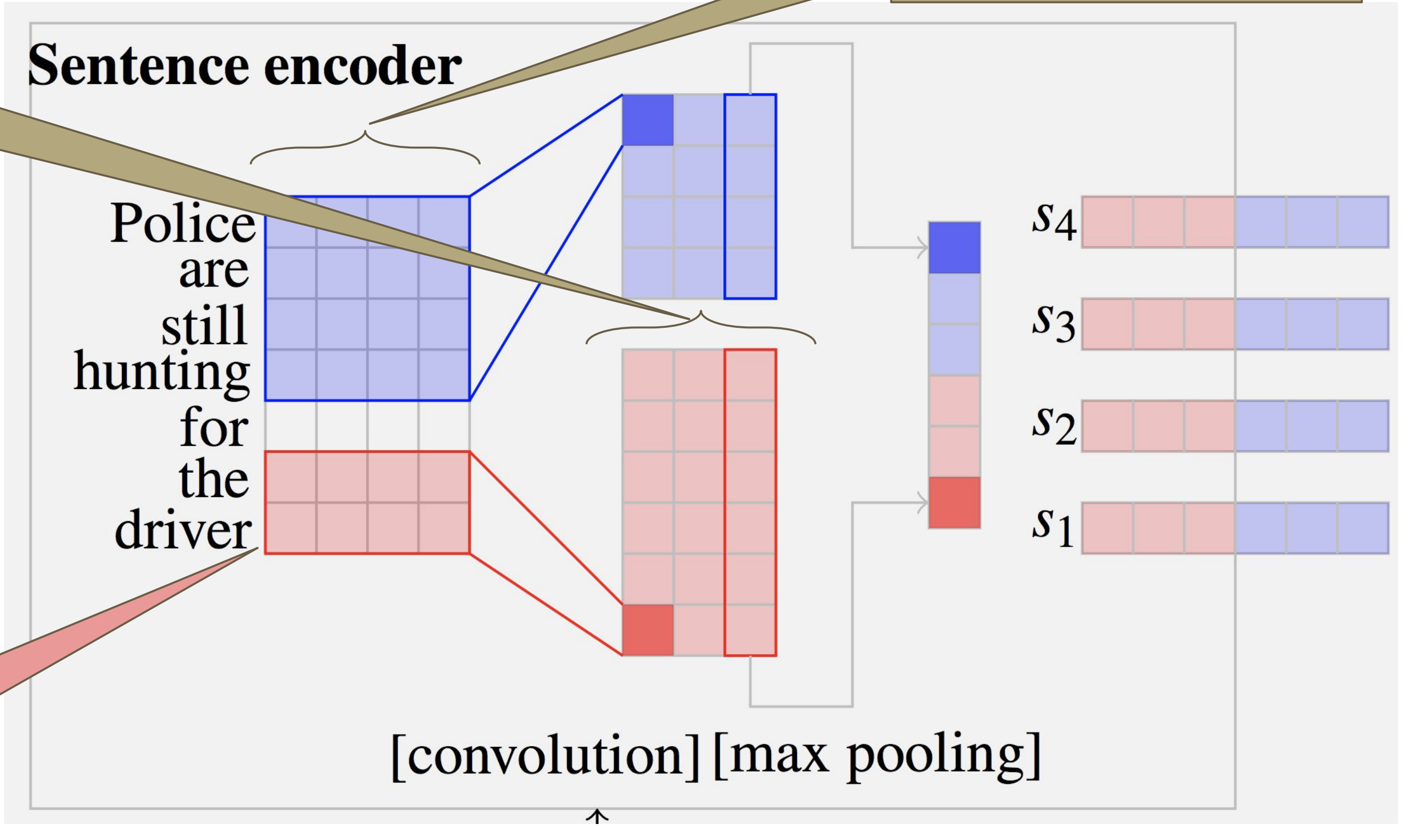


Convolutional Sentence Encoders

Word embeddings

Output channel width

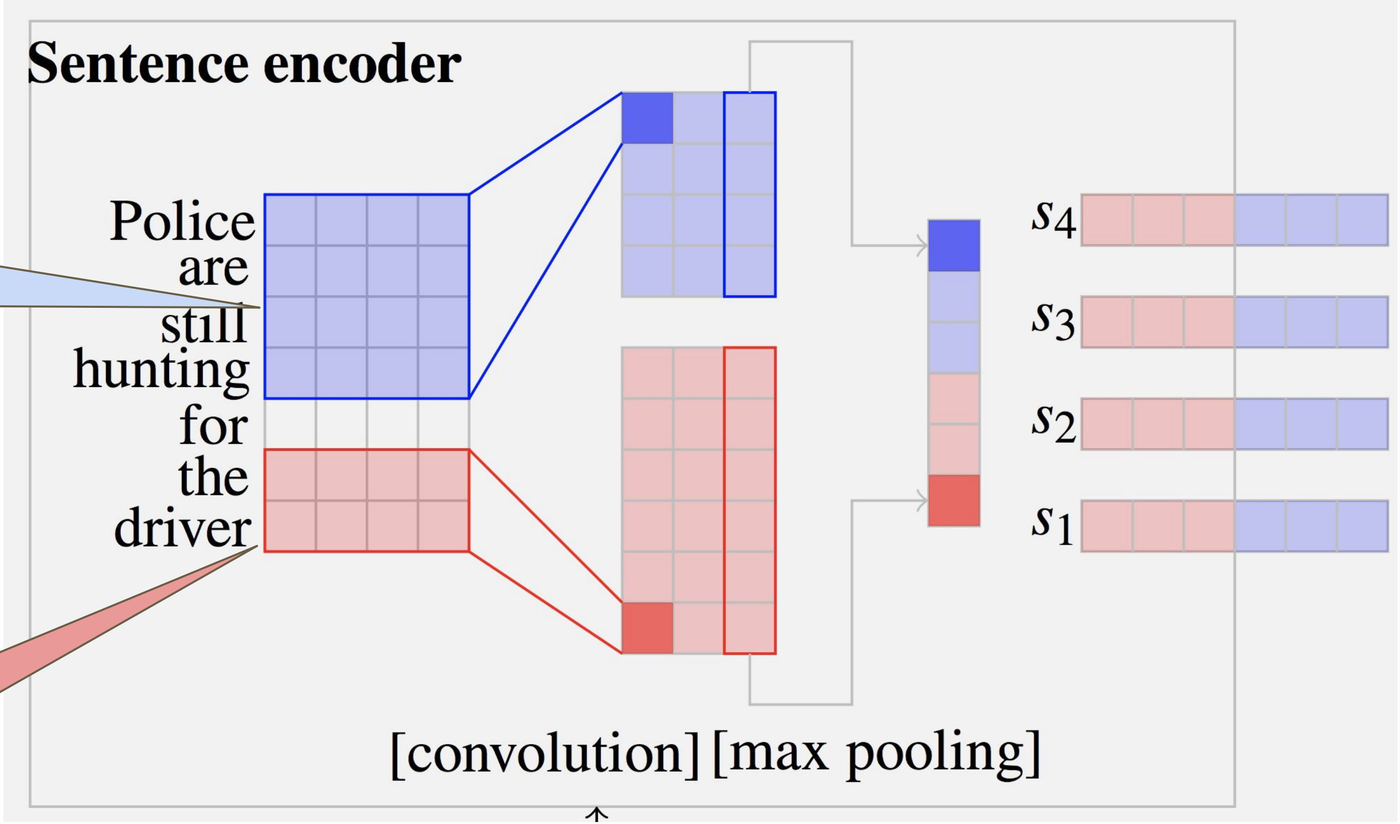
1-dimensional convolution (width 2)



Convolutional Sentence Encoders

1-dimensional convolution (width 4)

1-dimensional convolution (width 2)

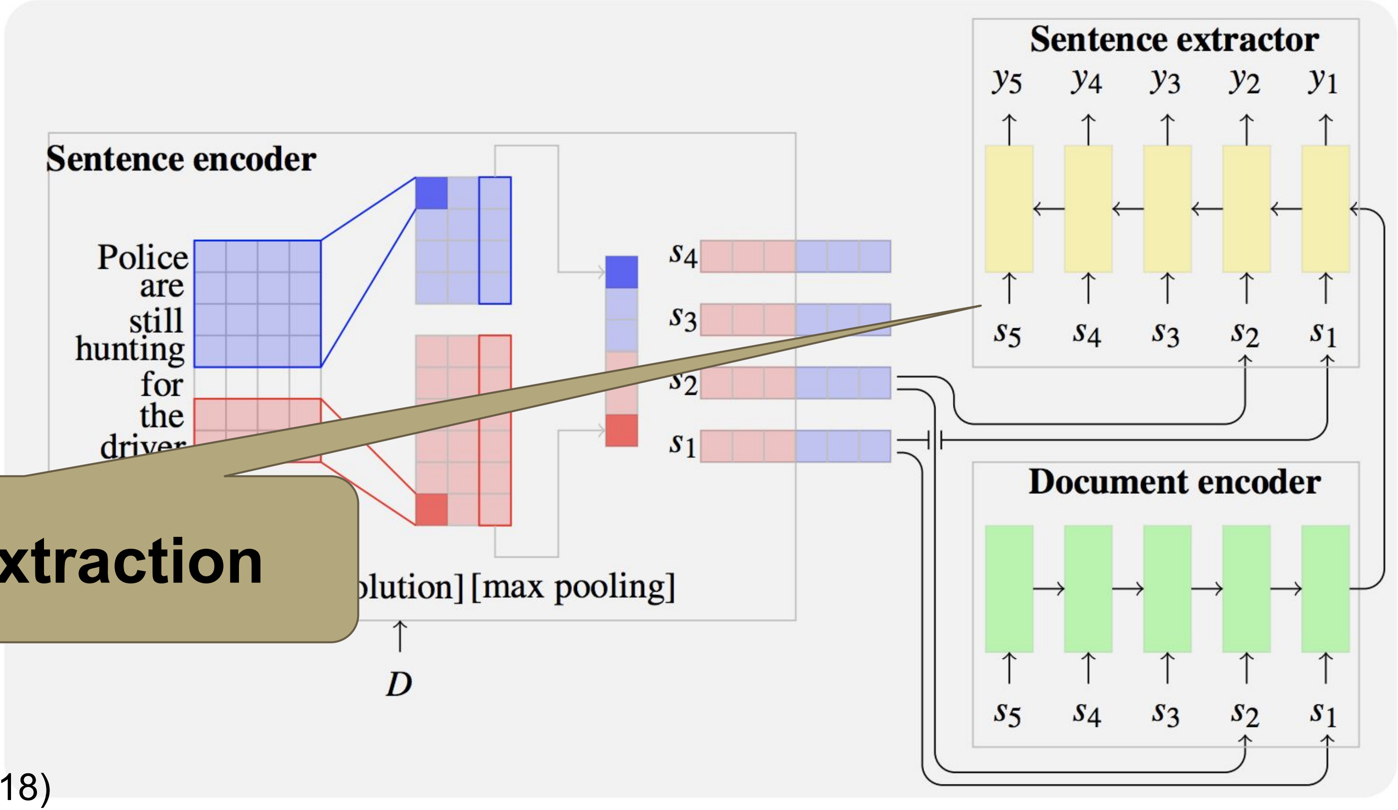


Convolutional Sentence Encoders

- ✓ Suited for document summarization for capturing salient information
- ✓ Issues with long range dependencies reduced
- ✗ Not clear how to utilize convolutional sentence encoders for abstractive summarization

Extractive Summarization

$$p(\ell|D;\theta), \text{ where } \ell \in \{0,1\}$$



Sentence Extraction

(Narayan et al., NAACL 2018)

Extractive Summarization with Convolutional Sentence Encoders

✓ Model achieves state-of-the-art performance for extractive summarization

Extractive Document Summarization (Narayan et al., NAACL 2018)

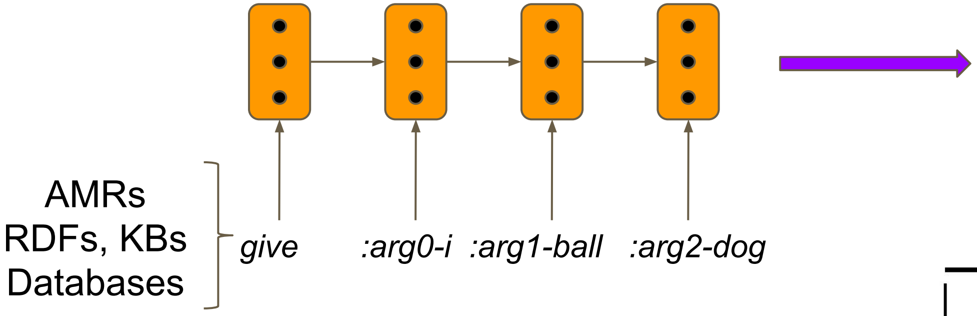
Extractive Summarization with Convolutional Sentence Encoders

- Model achi… state-of… nce f… extractive summarizati…

POSTER on Monday!

Extractive Document Summarization (Narayan et al., NAACL 2018)

Modeling Graph as a sequence of Tokens



Generation

AMR

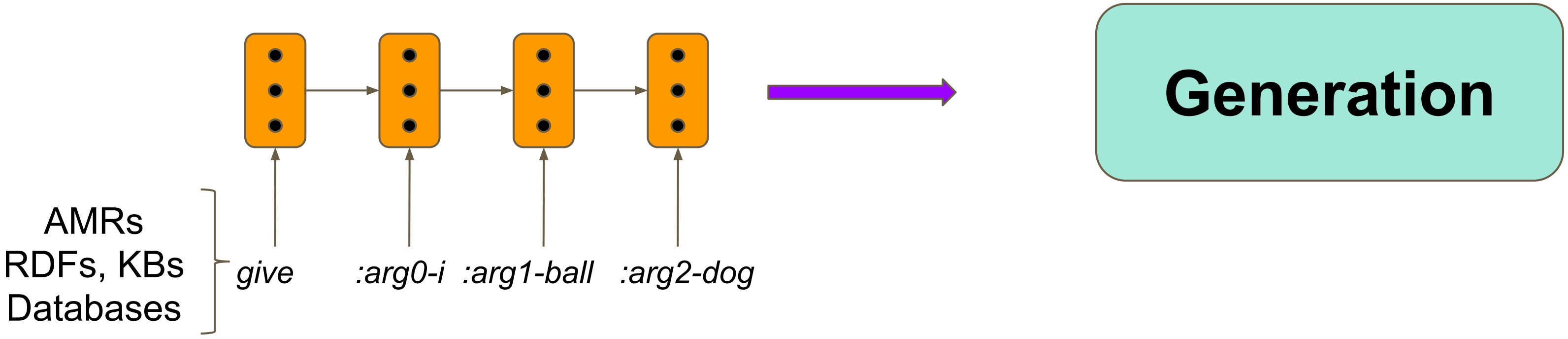
```
(g / give-01
   :ARG0 (i / I)
   :ARG1 (b / ball)
   :ARG2 (d / dog))
```

```
give :arg0 i :arg1 ball :arg2 dog
```

I $[\text{gave}]_{VP}$ $[\text{the dog}]_{NP}$ $[\text{a ball}]_{NP}$

I $[\text{gave}]_{VP}$ $[\text{the ball}]_{NP}$ $[\text{to a dog}]_{PP}$

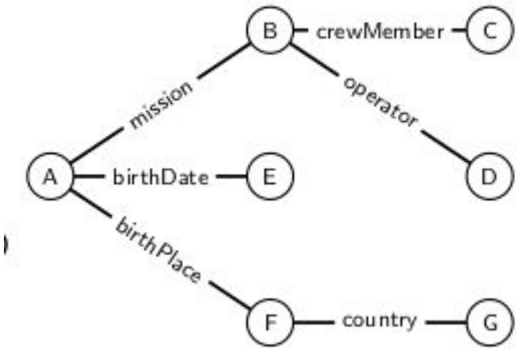
Modeling Graph as a sequence of Tokens



- **AMR Generation** (Konstas et al, 2017, Cao and Clark, 2018)
- **RDF Generation** (The WebNLG Challenge, Gardent et al. 2017)

Modeling Graphs as Sequence of Tokens

D2T Generation (Data = RDF)



LINEARISATION

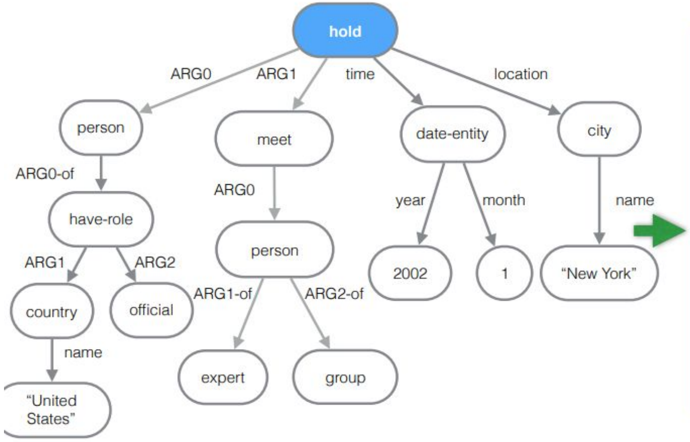
Alan_Bean **mission** Apollo_12 Apollo_12 **crewMember** Peter_Conrad Apollo_12 **operator** Nasa Alan_Bean birthDate 1932-03-15 Alan_Bean birthPlace Wheeler_Texas Wheeler_Texas country USA

Creating Training Corpora for NLG Micro-Planners (Gardent et al. 2017)

Modeling Graphs as Sequence of Tokens

MR-to-Text Generation

(MR = Abstract Meaning Representations)




```
hold
    :ARG0 (person
              :ARG0-of (have-role
                  :ARG1 United_States
                  :ARG2 official)
              )
    :ARG1 (meet
              :ARG0 (person
                  :ARG1-of expert
                  :ARG2-of group)
              )
    :time (date-entity 2002 1)
    :location New_York
```

US officials held an expert group meeting in January 2002 in New York .

Image from : Neural **AMR**: Sequence-to-Sequence Models for Parsing and **Generation** (Konstas et al, 2017)

Problems with Graph Linearization

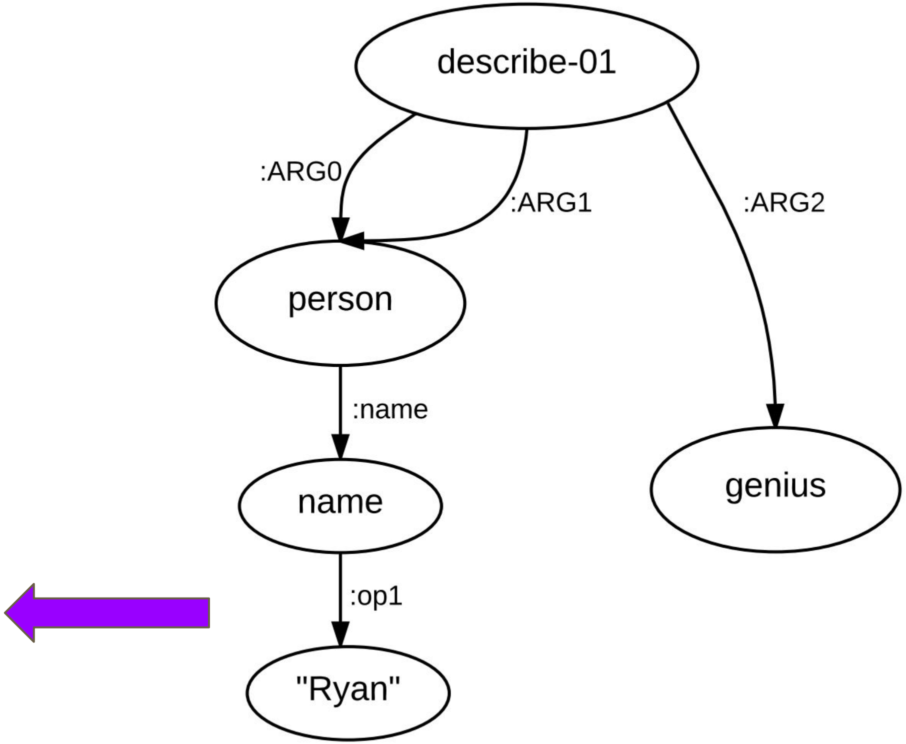
- ❌ Local dependencies available in the input turned into long-range dependencies
- ❌ RNNs often have trouble modeling long-range dependencies

Modeling with Graph Encoders

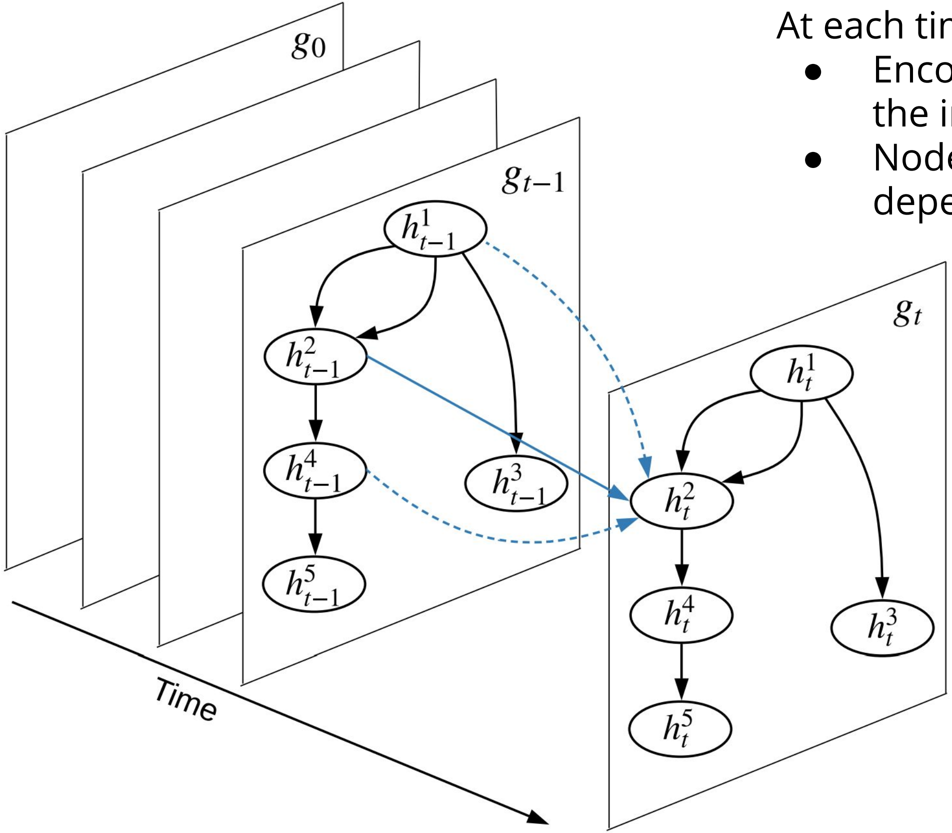
- ✓ **AMR Generation:** A Graph-to-Sequence Model for AMR-to-Text Generation (Song et al., ACL 2018)
- ✓ **RDF Generation:** GTR-LSTM: A Triple Encoder for Sentence Generation from RDF Data (Trisedya et al. ACL 2018)
- ✓ **Graph Convolutional Networks for SRL and NMT** (Kipf and Welling 2017, Marcheggiani and Titov, 2017, Bastings et al. 2017)

Graph-to-Sequence Model for AMR Generation

Ryan's description of himself: a genius.

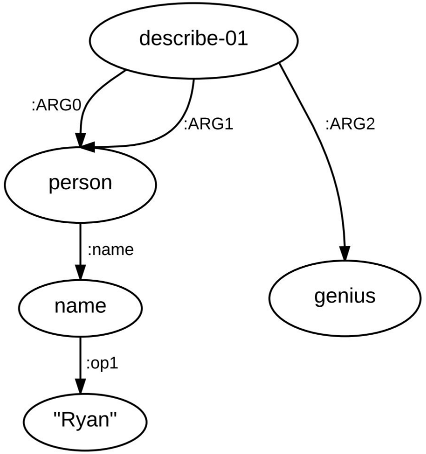


Graph-to-Sequence Model for AMR Generation



At each time step:

- Encoder operates directly on the graph structure of the input
- Node representations are updated using their dependents in the graph



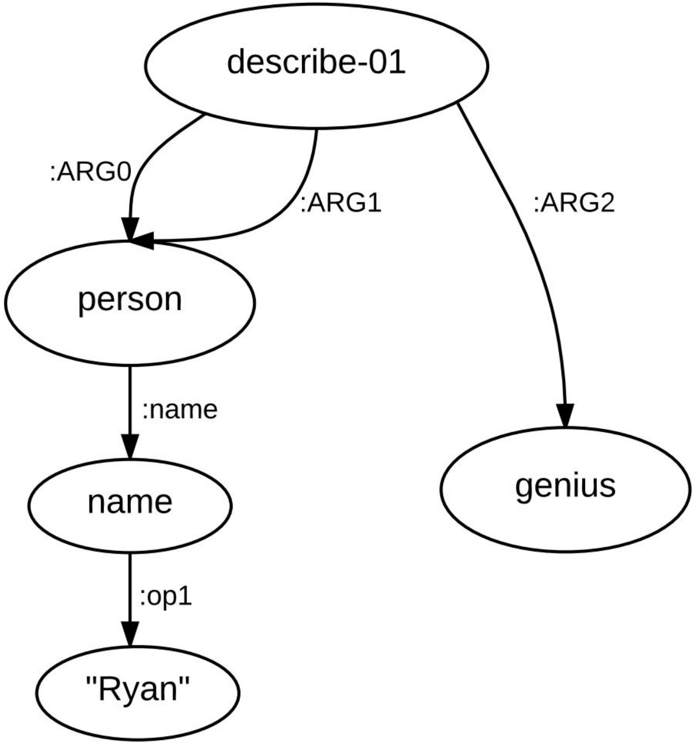
Graph-to-Sequence Model for AMR Generation

For node v_j, we define incoming and outgoing input representations:

$$x_j^i = \sum_{(i,j,l)\in E_{in}(j)} x_{i,j}^l$$

$$x_j^o = \sum_{(j,k,l)\in E_{out}(j)} x_{j,k}^l,$$

$x_{i,j}^l$ is an input representation for edge (i, j, l).

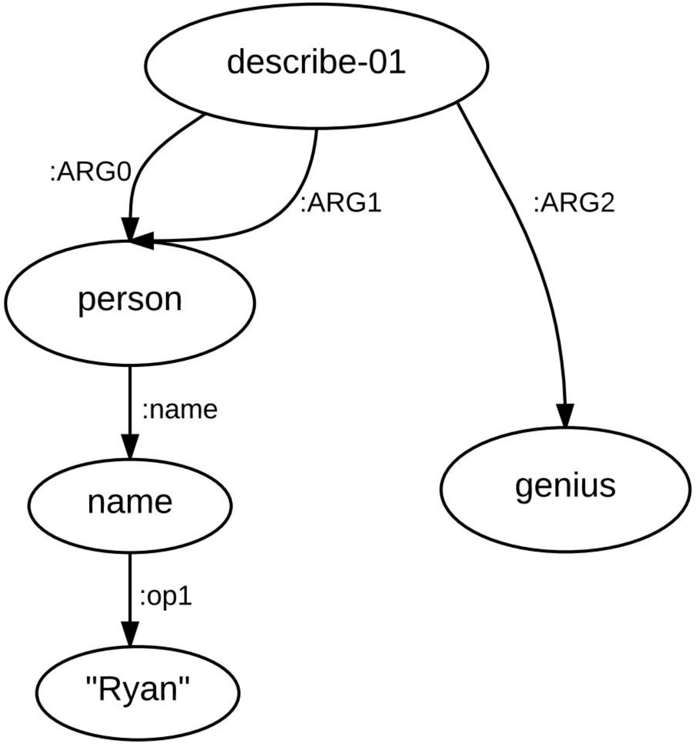


Graph-to-Sequence Model for AMR Generation

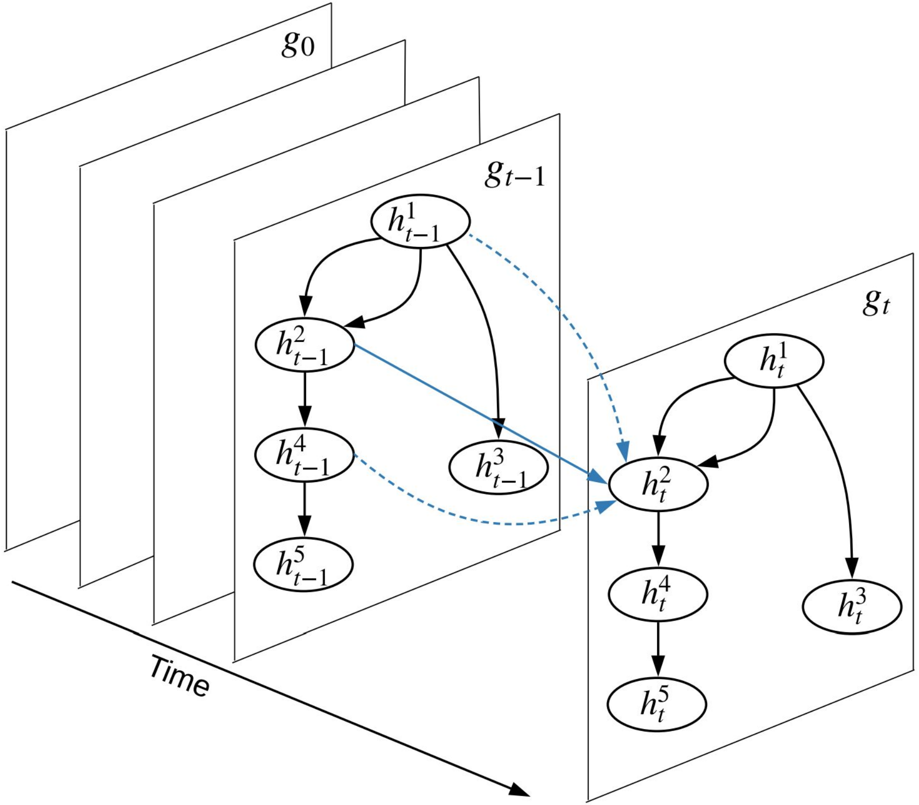
For node v_j, we define incoming and outgoing hidden state representations:

$$h_j^i = \sum_{(i,j,l) \in E_{in}(j)} h_{t-1}^i$$

$$h_j^o = \sum_{(j,k,l) \in E_{out}(j)} h_{t-1}^k,$$



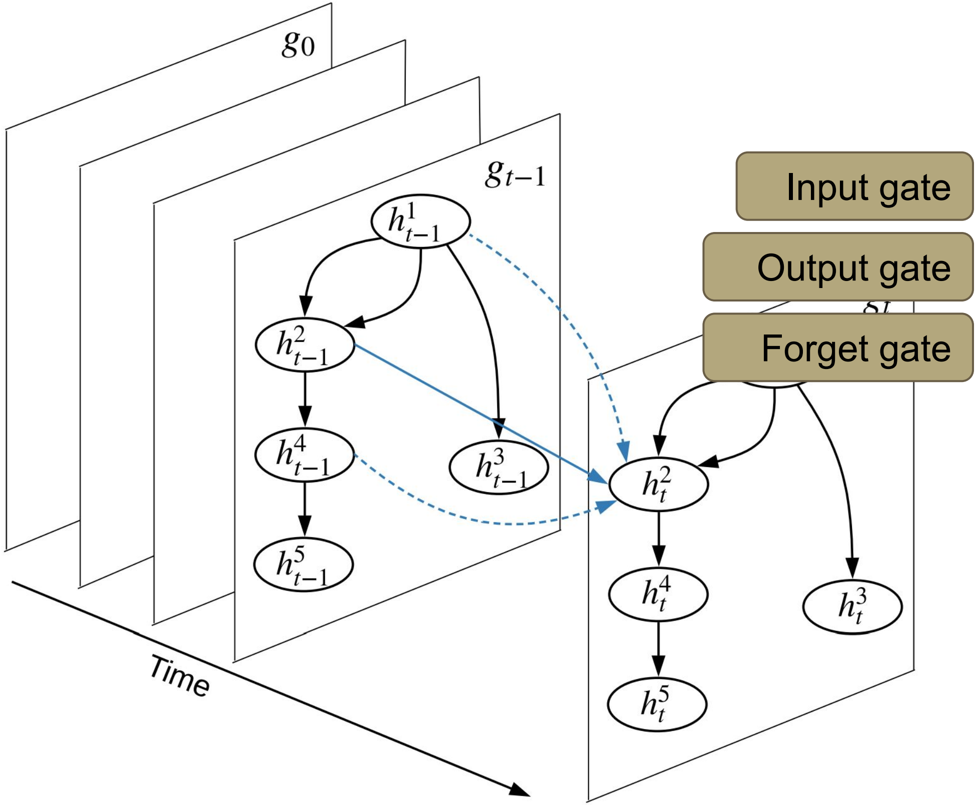
Graph-to-Sequence Model for AMR Generation



Graph state transition

$$\begin{aligned}
i_t^j &= \sigma(W_i x_j^i + \hat{W}_i x_j^o + U_i h_j^i + \hat{U}_i h_j^o + b_i), \\
o_t^j &= \sigma(W_o x_j^i + \hat{W}_o x_j^o + U_o h_j^i + \hat{U}_o h_j^o + b_o), \\
f_t^j &= \sigma(W_f x_j^i + \hat{W}_f x_j^o + U_f h_j^i + \hat{U}_f h_j^o + b_f), \\
u_t^j &= \sigma(W_u x_j^i + \hat{W}_u x_j^o + U_u h_j^i + \hat{U}_u h_j^o + b_u), \\
c_t^j &= f_t^j \odot c_{t-1}^j + i_t^j \odot u_t^j, \\
h_t^j &= o_t^j \odot \tanh(c_t^j),
\end{aligned}$$

Graph-to-Sequence Model for AMR Generation

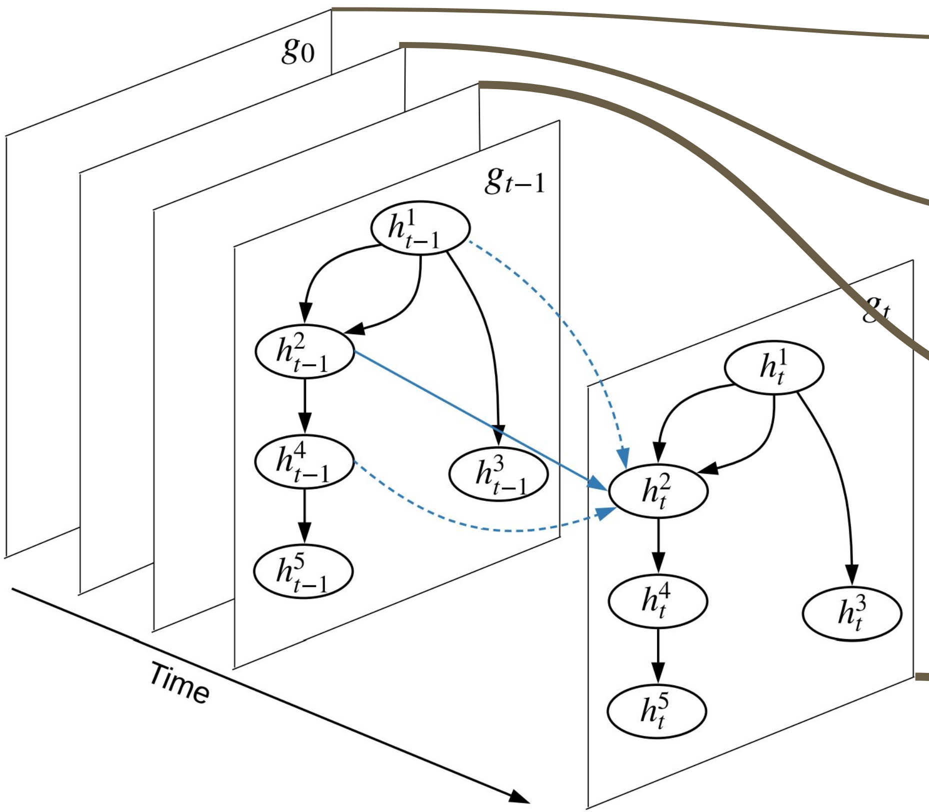


Graph state transitions

$$
\begin{aligned}
i_t^j &= \sigma(W_i x_j^i + \hat{W}_i x_j^o + U_i h_j^i + \hat{U}_i h_j^o + b_i),\\
o_t^j &= \sigma(W_o x_j^i + \hat{W}_o x_j^o + U_o h_j^i + \hat{U}_o h_j^o + b_o),\\
f_t^j &= \sigma(W_f x_j^i + \hat{W}_f x_j^o + U_f h_j^i + \hat{U}_f h_j^o + b_f),\\
u_t^j &= \sigma(W_u x_j^i + \hat{W}_u x_j^o + U_u h_j^i + \hat{U}_u h_j^o + b_u),\\
c_t^j &= f_t^j \odot c_{t-1}^j + i_t^j \odot u_t^j,\\
h_t^j &= o_t^j \odot \tanh(c_t^j),
\end{aligned}
$$

LSTM

Graph-to-Sequence Model for AMR Generation



t=0: Represents each node itself

t=1: Represents nodes with their immediate parents and children

t=2: Represents nodes with their grandparents and grandchildren

t=longest path length: Represents nodes with whole graph knowledge

Decoding with Graph-to-Sequence Model

- Standard attention-based LSTM decoder
- **Decoder initial state**: Average of the last states of all nodes
- Standard **attention** and **copy mechanism** can be used on the last states of all nodes

Graph-to-Sequence Model for AMR Generation

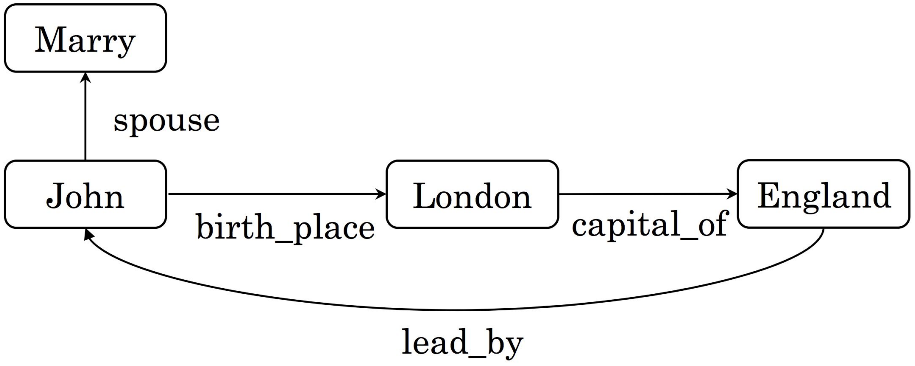
✓ **Model achieves state-of-the-art performance outperforming (Konstas et al. 2017) for AMR Generation.**

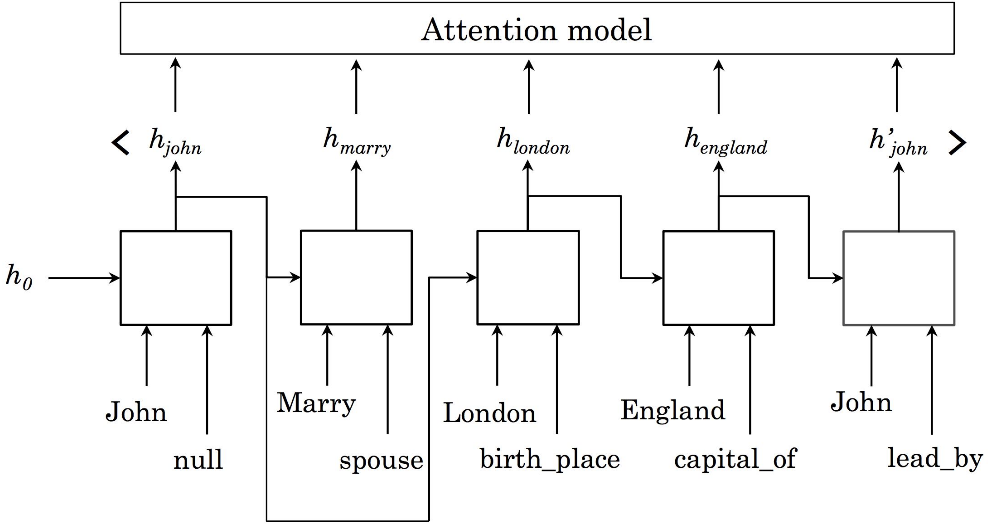
A Graph-to-Sequence Model for AMR-to-Text Generation (Song et al., 2018)

Graph-based Triple Encoder for RDF Generation

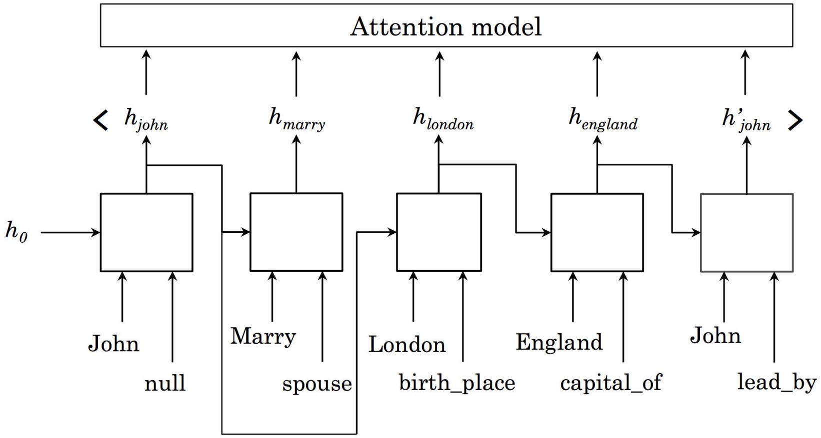
RDF triples	⟨John Doe,birth place,London⟩ ⟨John Doe,birth date,1967-01-10⟩ ⟨London,capital of,England⟩
Target sentence	John Doe was born on 1967-01-10 in London, the capital of England.

Graph-based Triple Encoder for RDF Generation

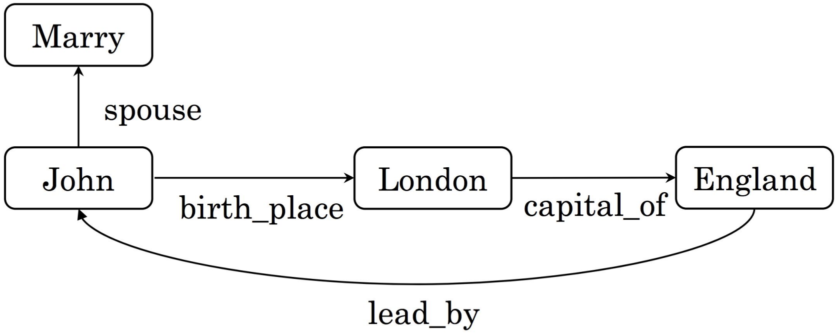




Graph-based Triple Encoder for RDF Generation

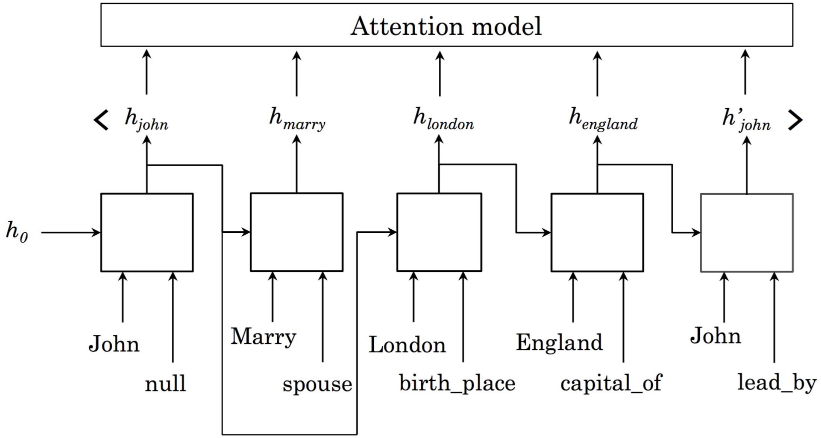


- Traverses the input graph
- When a vertex is visited, the hidden states of adjacent vertices are created
- Each GTR-LSTM unit receives an entity and the incoming property



Graph-based Triple Encoder for RDF Generation

$$
\begin{aligned}
i_t &= \sigma\left(\sum_e (U^{ie} x_{te} + W^{ie} h_{t-1})\right) \\
f_{te} &= \sigma\left(U^f x_{te} + W^f h_{t-1}\right) \\
o_t &= \sigma\left(\sum_e (U^{oe} x_{te} + W^{oe} h_{t-1})\right) \\
g_t &= \tanh\left(\sum_e (U^{ge} x_{te} + W^{ge} h_{t-1})\right) \\
c_t &= \left(c_{t-1} * \sum_e f_{te}\right) + (g_t * i_t) \\
h_t &= \tanh(c_t) * o_t
\end{aligned}
$$



Separate forget gate for each input to allow GTR-LSTM unit to incorporate information from each input selectively

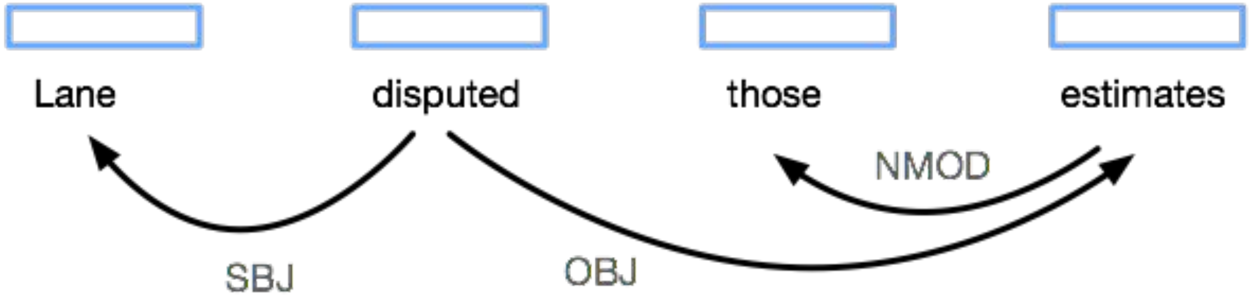
GTR-LSTM: Triple Encoder for RDF Generation

✓ **Model holds current state-of-the-art performance for RDF Generation on the WebNLG Dataset**

GTR-LSTM: A Triple Encoder for Sentence Generation from RDF Data. (Trisedya et al. ACL 2018)

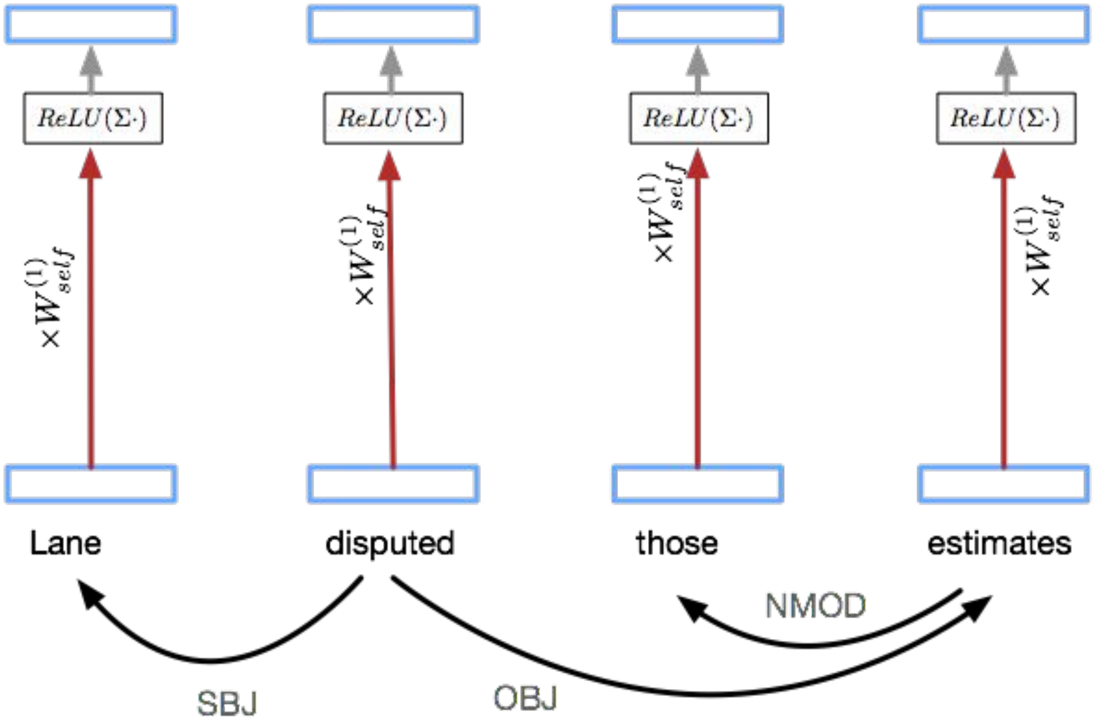
Graph Convolutional Networks

Encoding dependency structures for SRL and NMT

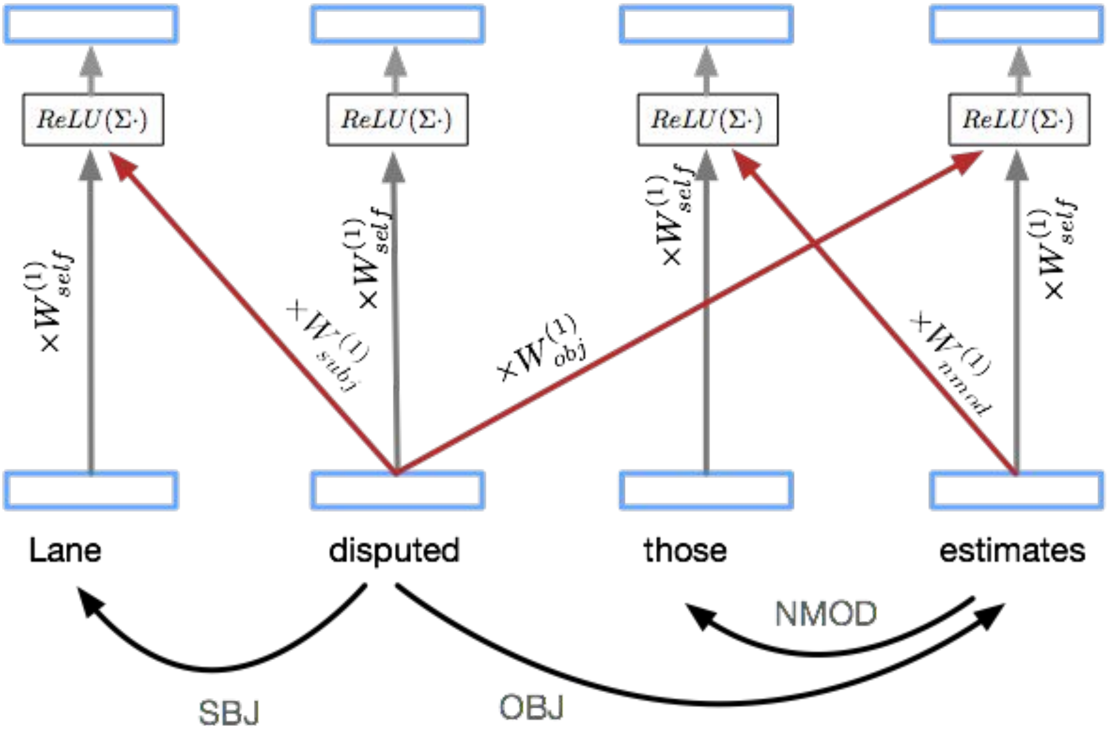


(Kipf & Welling 2017)

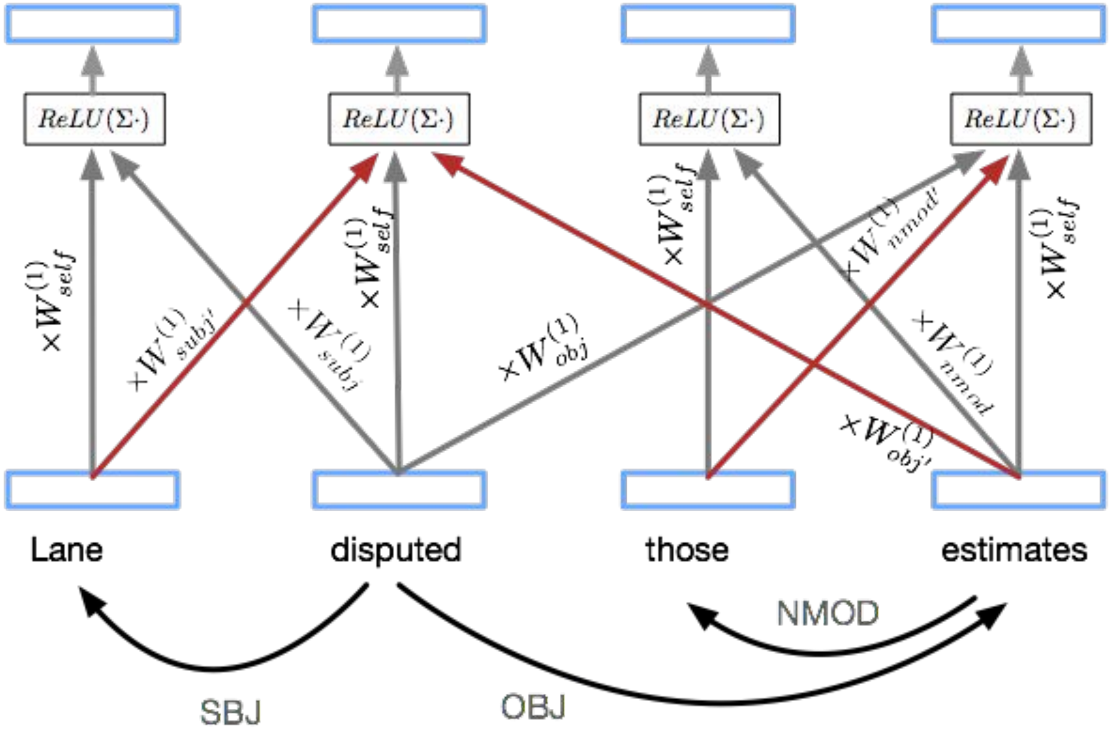
Graph Convolutional Networks



Graph Convolutional Networks

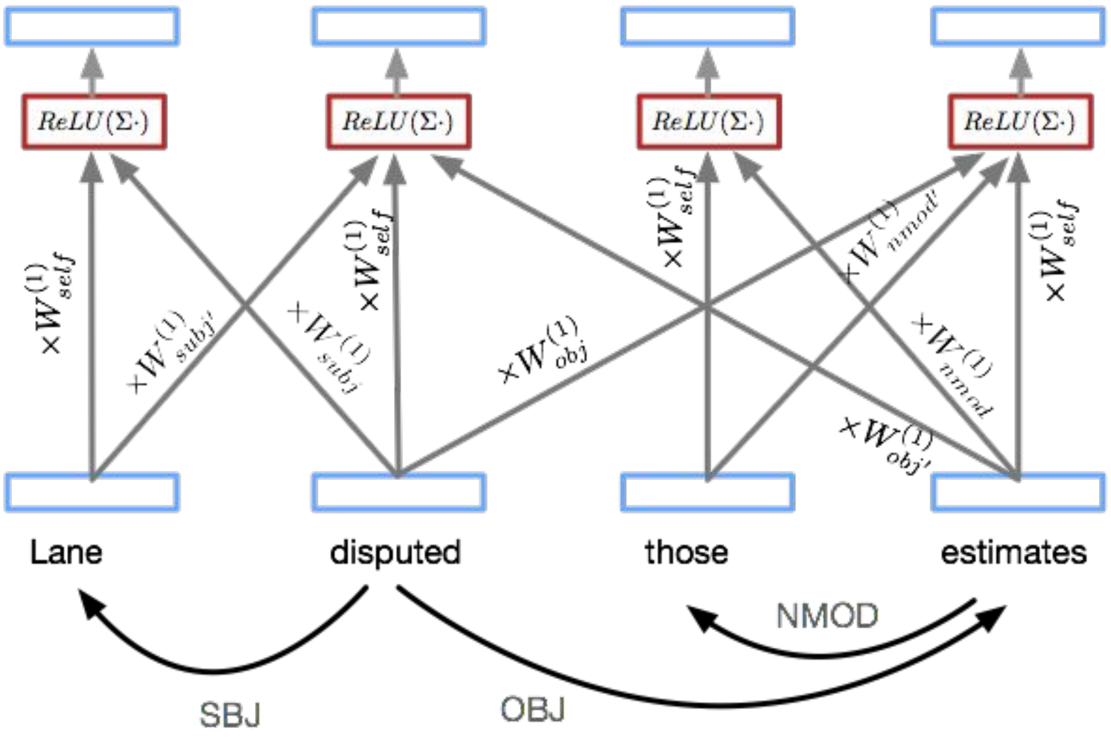


Graph Convolutional Networks



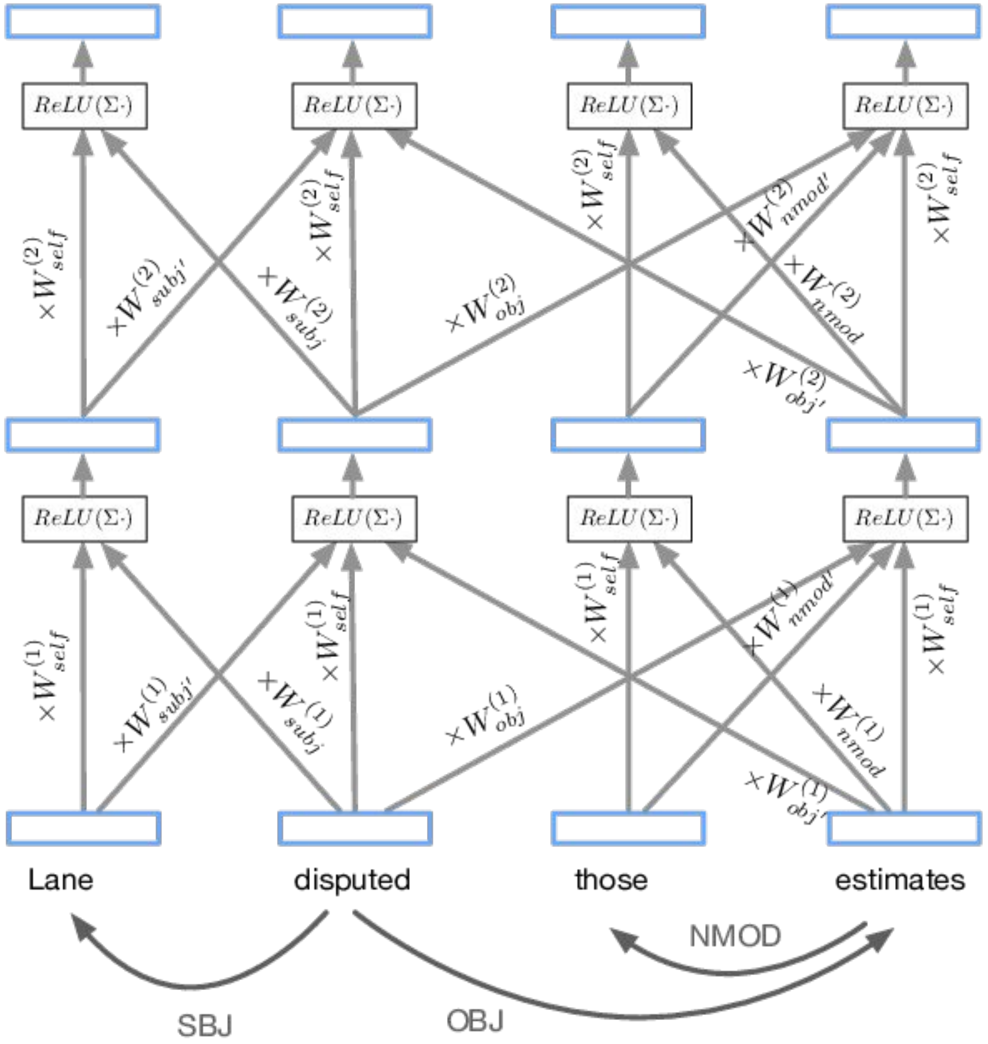
Graph Convolutional Networks

$$h_v = ReLU\left(\sum_{u \in \mathcal{N}(v)} (Wx_u + b)\right)$$



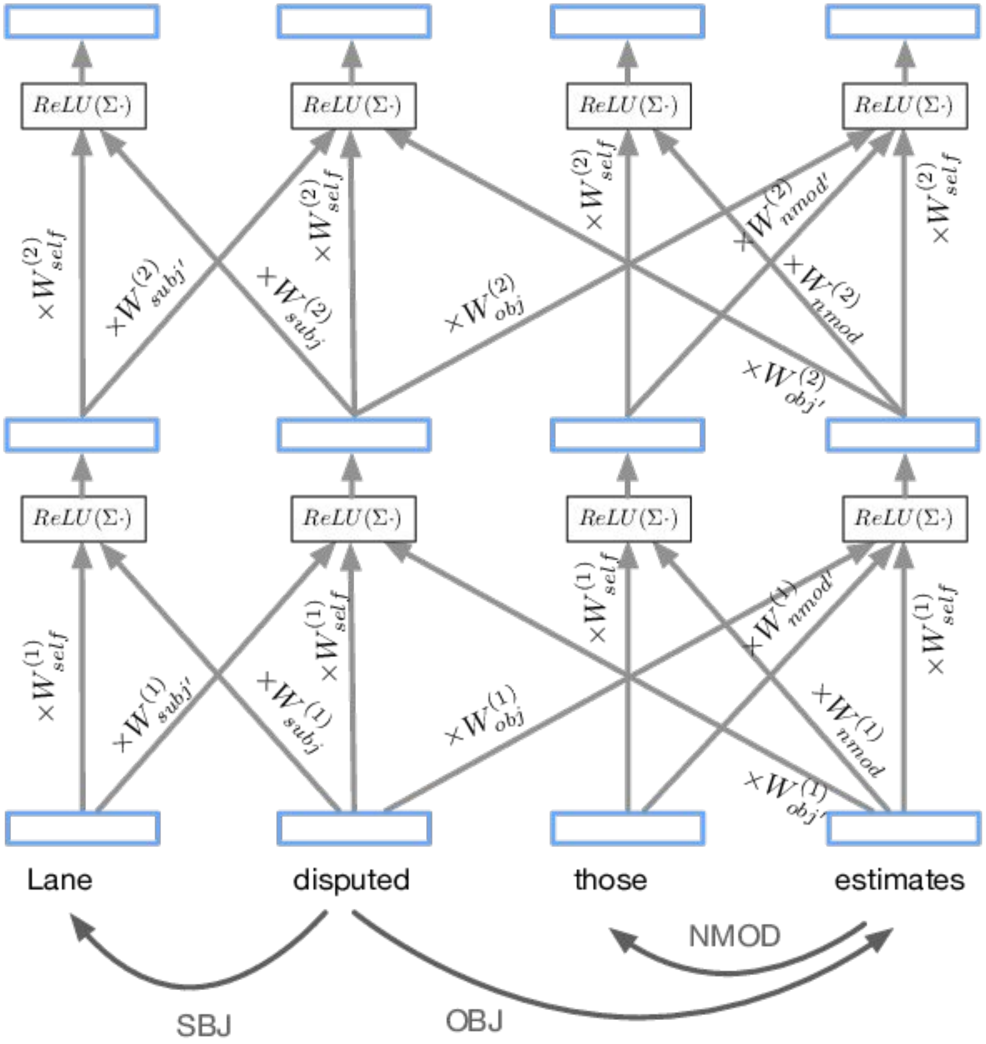
Graph Convolutional Networks

$$h_v^{(k+1)} = ReLU\left(\sum_{u \in \mathcal{N}(v)} W^{(k)} h_u^{(k)} + b^{(k)}\right)$$



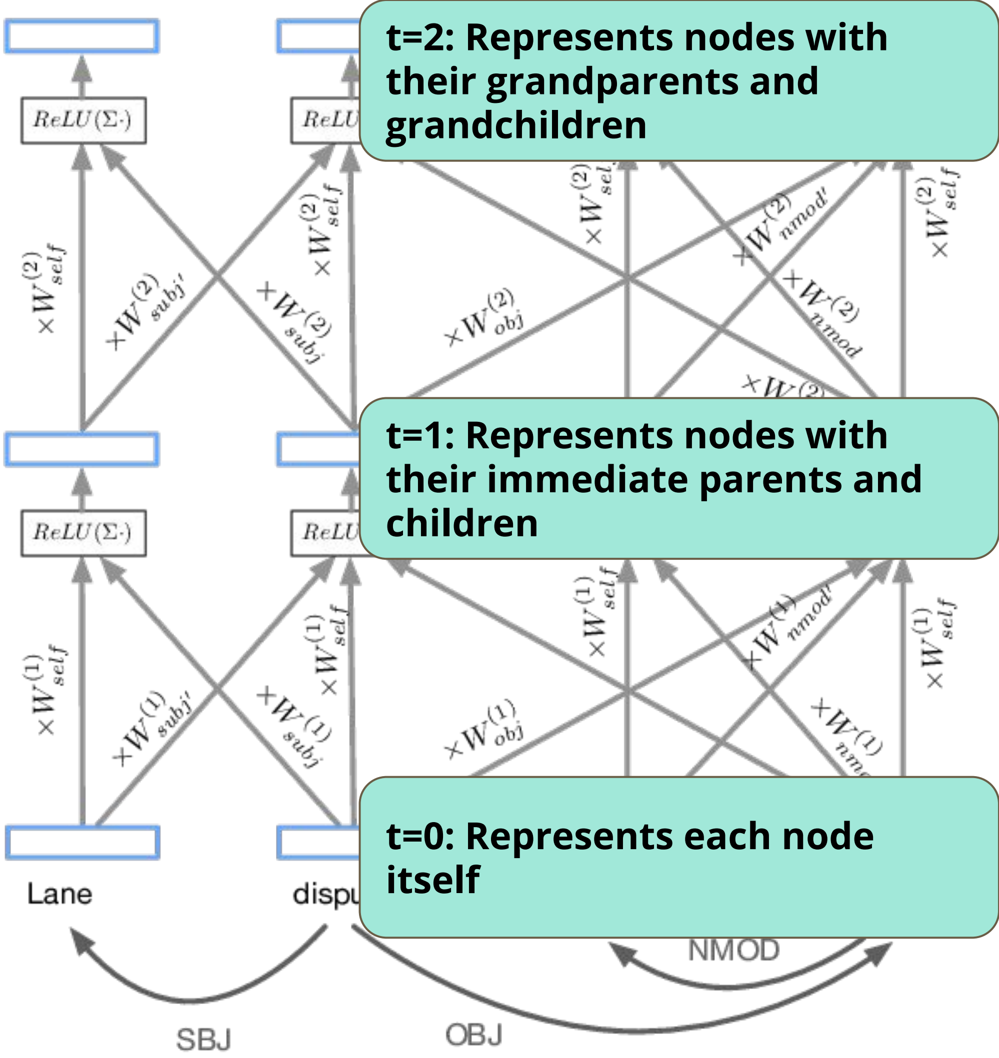
Graph Convolutional Networks

Similarity with Graph-to-Sequence Models



Graph Convolutional Networks

Similarity with Graph-to-Sequence Models



Graph Convolutional Networks

- **Semantic Role Labeling**: (Marcheggiani and Titov 2017).
- Syntax-aware **Neural Machine Translation:** (Bastings et al. 2017)
- **AMR or RDF Generation?**

(Kipf & Welling, 2017)

Summary: Input Representation and Text Production

Hierarchical document encoders and graph encoders are able to better model input for text production

State of art results on Summarization, AMR generation and RDF generation

Many more to come...

Communication Goal-Oriented Deep Generators

Communication Goal-Oriented Deep Generators

✓ Infusing task-specific knowledge to deep architectures

✓ Reinforcement Learning: Optimizing final evaluation metric

Infusing Task-Knowledge to Deep Architectures

Similarity with Machine Translation: Text production tasks such as paraphrase generation and AMR generation often have **semantic equivalence between source and target sides.**

However, it is not true for text production in general:

- Sentence Compression or Simplification
- Generation from noisy data
- Document Summarization
- Conversational Agents

Sentence Summarization, Sentence Compression or Title Generation

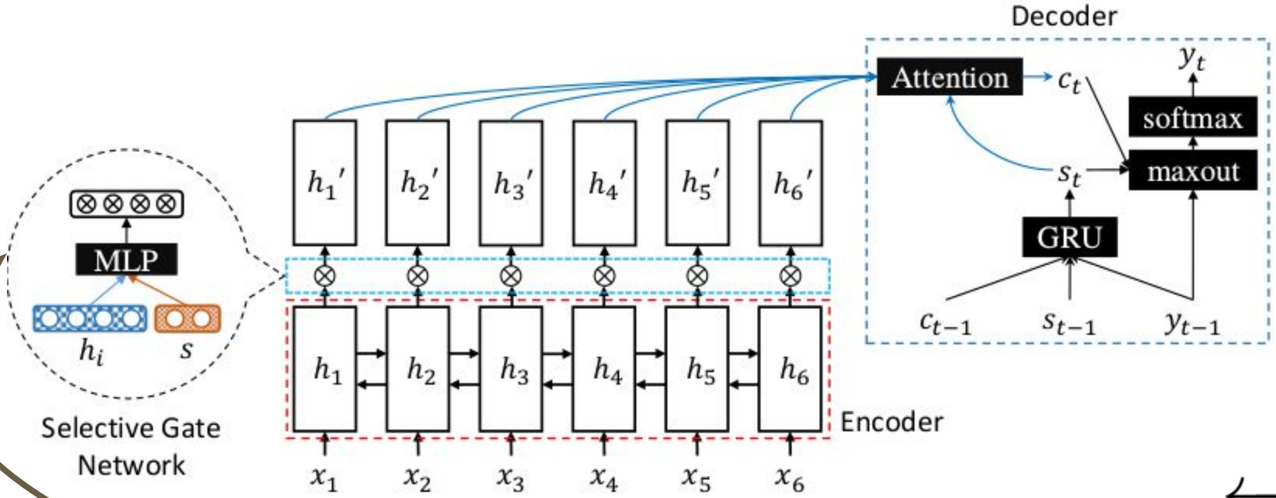
the **sri lankan** government on wednesday announced the **closure** of **government schools** with **immediate effect** as a **military campaign** against **tamil separatists escalated** in the north of the country .

↓

sri lanka closes schools as war escalates

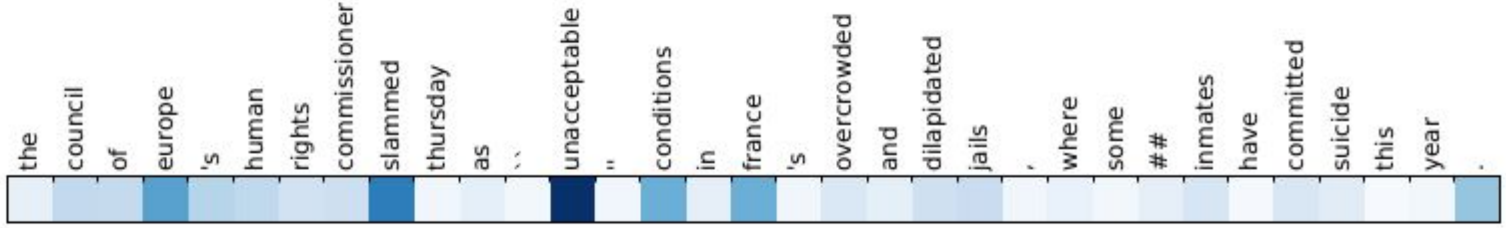
Abstractive Sentence Summarization (Zhou et al. 2017)

Selective Encoding to Capture Salient Information



$$
\begin{aligned}
s &= [\overleftarrow{h_1} ; \overrightarrow{h_n}] \\
sGate_i &= \sigma(W_s h_i + U_s s + b) \\
h_i^{'} &= h_i \odot sGate_i
\end{aligned}
$$

Selective Encoding to Capture Salient Information



Input: The Council of Europe's human rights commissioner slammed thursday as "unacceptable" conditions in France's overcrowded and dilapidated jails, where some ## inmates have committed suicide this year.

Output Summary: Council of Europe slams French prisons conditions

Reference summary: Council of Europe again slams French prisons conditions

Generation from Loosely Aligned Noisy Data

Born	Robert Joseph Flaherty February 16, 1884 Iron Mountain, Michigan, U.S.
Died	July 23, 1951 (aged 67) Dummerston, Vermont, U.S.
Cause of death	Cerebral thrombosis
Occupation	Filmmaker
Spouse(s)	Frances Johnson Hubbard

Robert Joseph Flaherty, (**February 16, 1884 July 23, 1951**) was an **American film-maker**. Flaherty was married to **Frances H. Flaherty** until his death in 1951.

(Perez-Beltrachini and Lapata, NAACL 2018)

Generation from Loosely Aligned Noisy Data

Born	Robert Joseph Flaherty February 16, 1884 ~~Iron Mountain, Michigan,~~ U.S.
Died	July 23, 1951 ~~(aged 67)~~ ~~Dummerston, Vermont,~~ U.S.
~~**Cause of death**~~	~~Cerebral thrombosis~~
Occupation	Filmmaker
Spouse(s)	Frances Johnson Hubbard

Robert Joseph Flaherty, (**February 16, 1884 July 23, 1951**) was an **American film-maker**. Flaherty was married to **Frances H. Flaherty** until his death in 1951.

(Perez-Beltrachini and Lapata, NAACL 2018)

Generation with Multi-task Objective

Word prediction (generation) objective: $$\mathcal{L}_{wNLL} = -\sum_{t=1}^{|Y|} log\, P(y_t | y_{1:t-1}, X)$$

Content selection objective: $$\mathcal{L}_{aln} = -\sum_{t=1}^{|Y|} log\, P(a_t | y_{1:t-1}, X)$$

Multi-Task Learning (Caruana, 1993): $$\mathcal{L}_{MTL} = \lambda\, \mathcal{L}_{wNLL} + (1 - \lambda)\, \mathcal{L}_{aln}$$

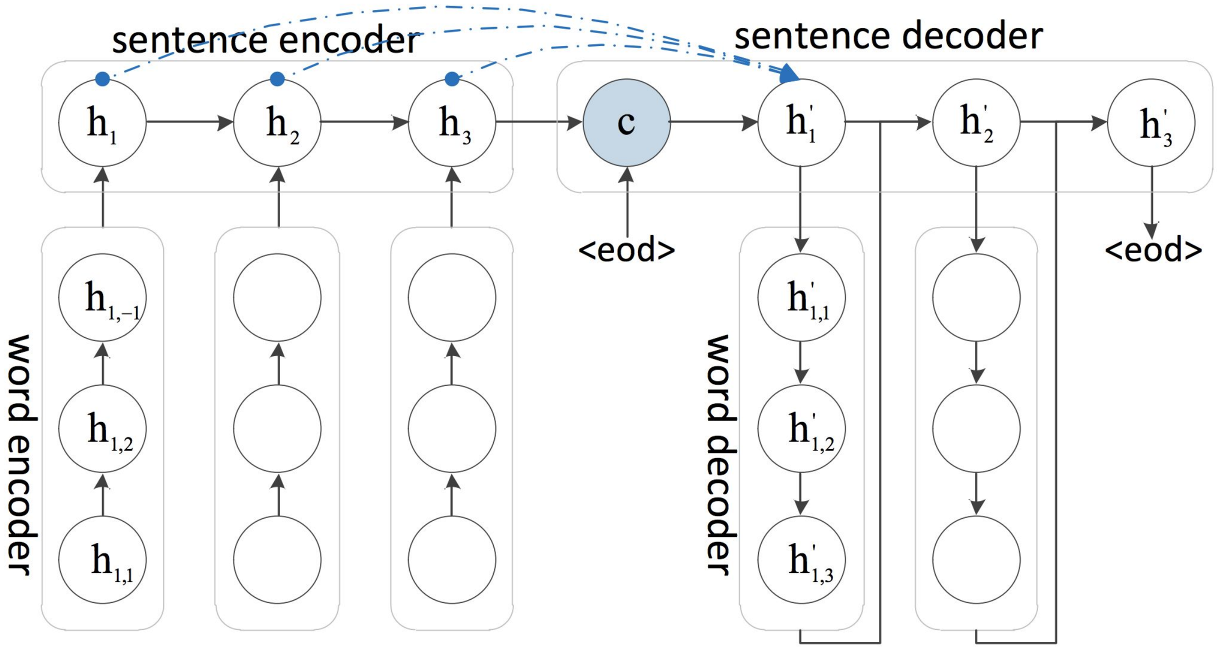
Generation from Loosely Aligned Noisy Data

Experimental results show that **models trained with content-specific objectives** **improve** upon **vanilla encoder-decoder architectures** which rely solely on soft attention

(Perez-Beltrachini and Lapata, NAACL 2018)

Document Summarization

Summarization requires model to **distinguish the content** that is **relevant for the summary** from the content that is not

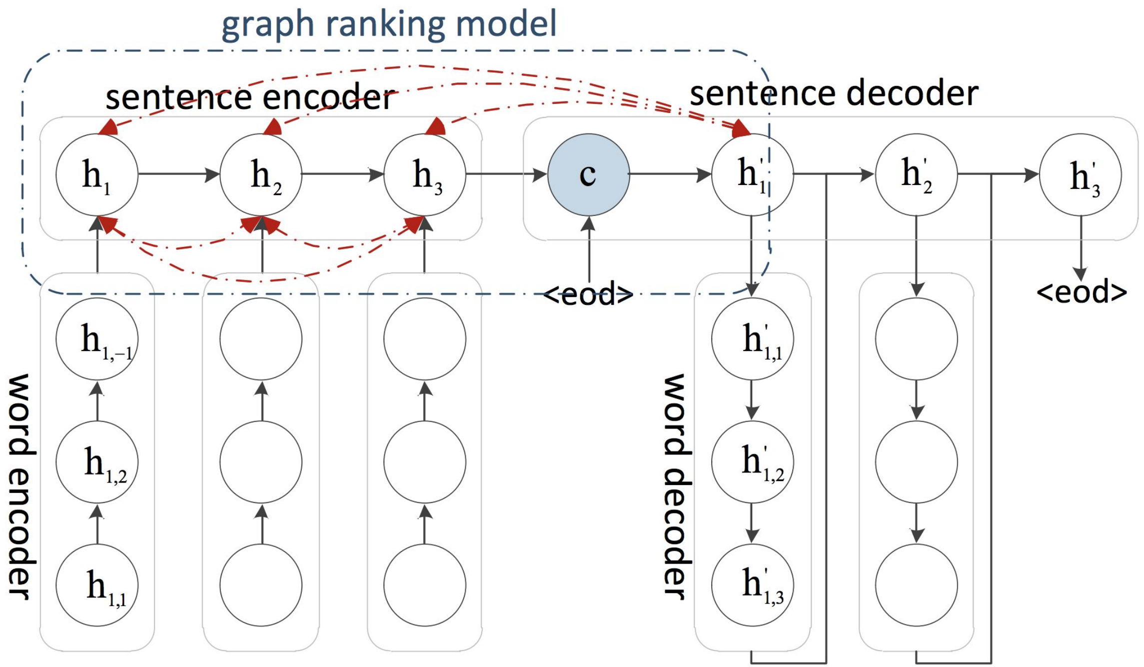


Abstractive Document Summarization (Tan et al. ACL 2017)

Document Summarization with Modified Attention

Identifying Salient sentences with topic-sensitive PageRank

A sentence is important in a document if it is heavily linked with many important sentences.



Abstractive Document Summarization (Tan et al. 2017)

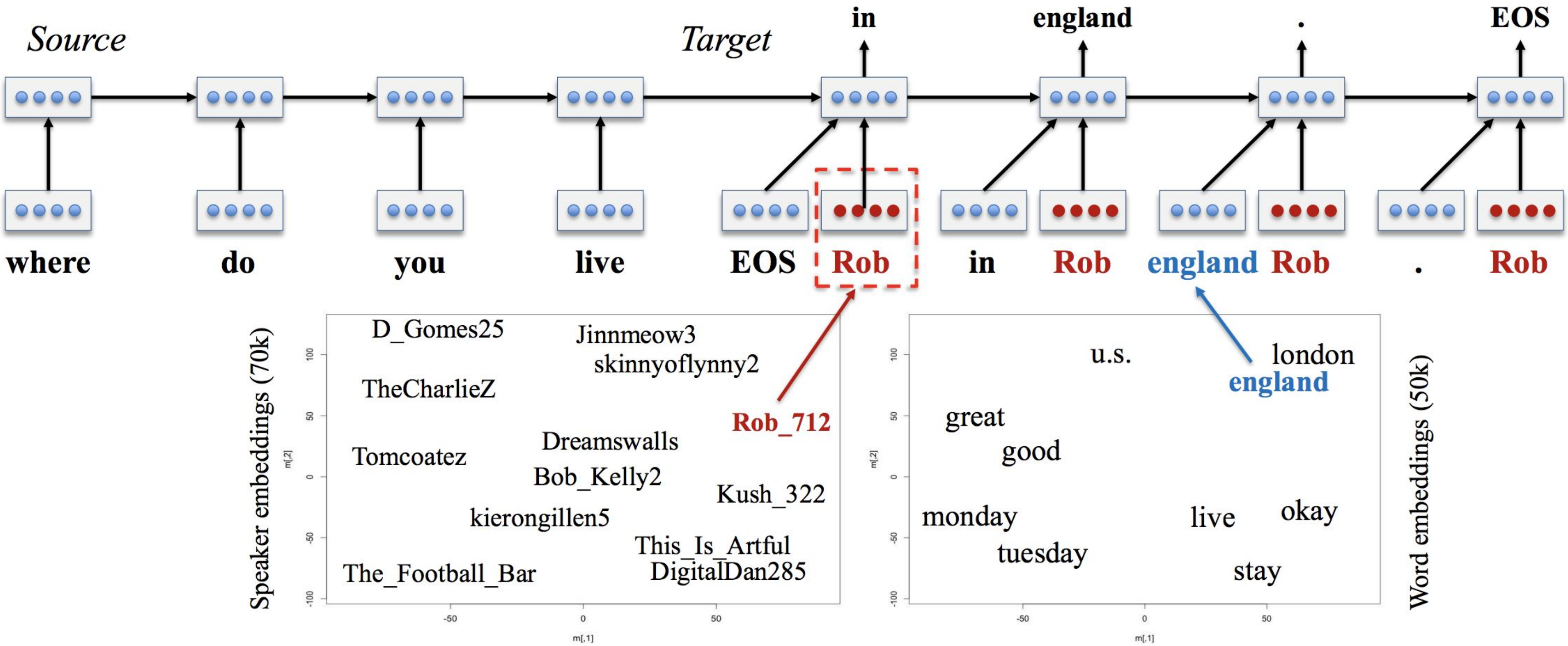
User-Profiling in Neural Conversational Model

User1	
message	Where is your hometown?
response	I’m from England.
message	Where are you from?
response	I’m from England.
message	In which city do you live now?
response	I live in London.
message	In which country do you live now?
response	I live in England.
message	What is your major?
response	Business. You?
message	What did you study in college?
response	I did business studies.
message	How old are you?
response	I’m 18.
message	What is your age?
response	I’m 18.

User2	
message	Where is your hometown?
response	I live in Indonesia.
message	Where are you from?
response	England, you?
message	In which city do you live now?
response	I live in Jakarta.
message	In which country do you live now?
response	I live in Indonesia.
message	What is your major?
response	Business, you?
message	What did you study in college?
response	Psychology, you?
message	How old are you?
response	I’m 18.
message	What is your age?
response	I’m 16.

A Persona-Based Neural Conversation Model (Li et al. 2016)

User-Profiling in Neural Conversational Model



Infusing Task-Knowledge to Deep Architectures

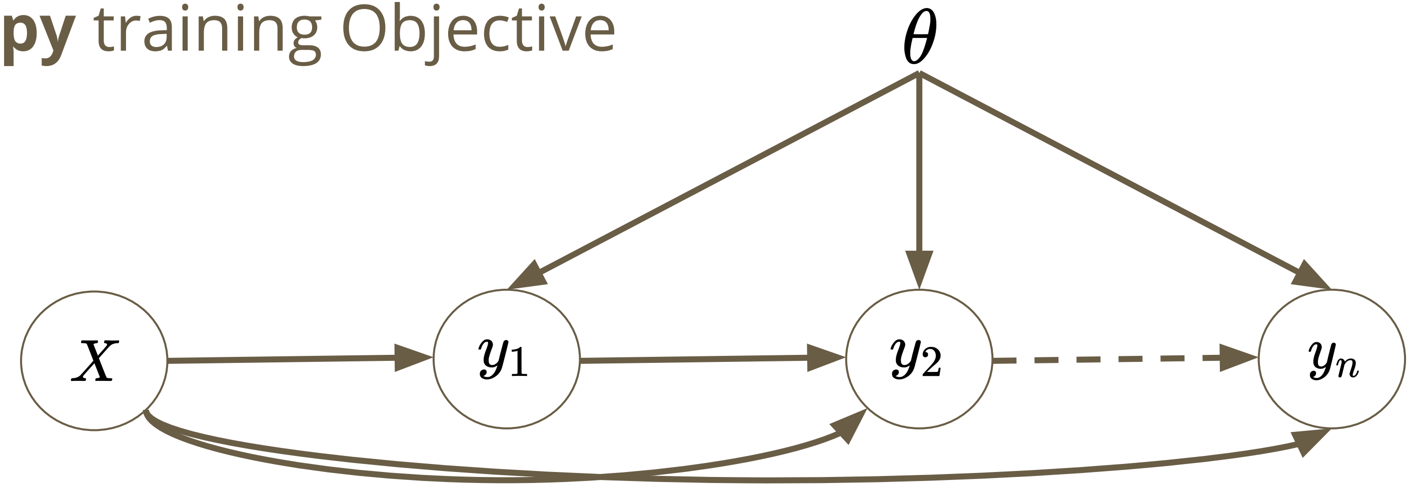
Is big data an alternative solution? Maybe!

Infusing task-knowledge reflecting communication goal immediately helps!!

Enforcing Model to Optimize Task-Specific Metric

$$L(\theta) = -\sum_{i=1}^{n} \log p(y_i | y_1, \ldots, y_{i-1}, X; \theta)$$

Cross-Entropy training Objective



Enforcing Model to Optimize Task-Specific Metric

$$L(\theta) = -\sum_{i=1}^{n} \log p(y_i | y_1, \ldots, y_{i-1}, X; \theta)$$

Cross-Entropy training Objective is **not Optimal**!

- It maximizes the **likelihood of the next correct word** and **not the task-specific evaluation metrics**.
- In addition, it suffers from **exposure bias problem**.

Enforcing Model to Optimize Task-Specific Metric

Automatic Evaluation

- AMR Generation: BLEU, SMATCH
- RDF and Dialog Act Generation: BLEU, NIST, METEOR etc.
- End-to-end Dialog Models: BLEU
- Sentence Compression and Simplification: BLEU, Compression Rate, SARI, Readability etc.
- Summarization: ROUGE, Pyramid

Human Evaluation

- Grammaticality, Fluency, Discourse Coherency, Preference, Informativeness etc.

Exposure Bias with Cross Entropy Training

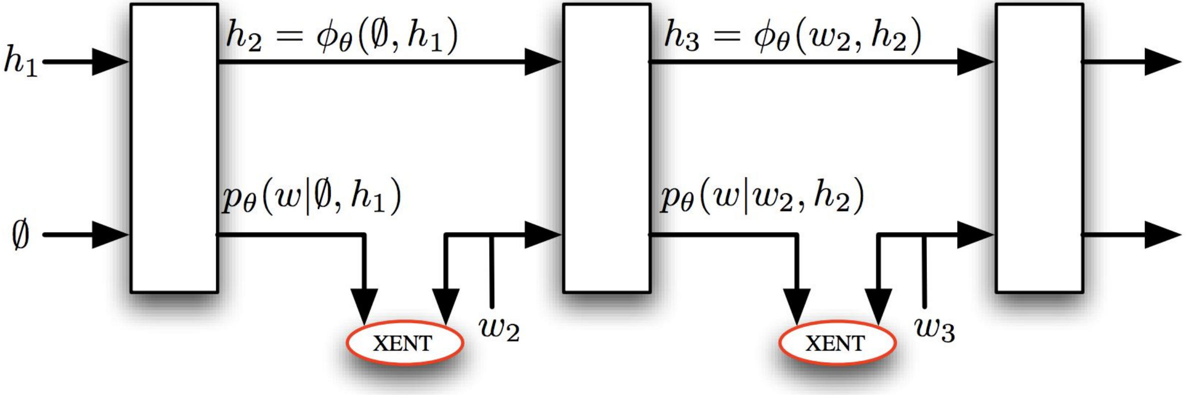
Training

Predict the next word in a sequence, given the previous reference words and context

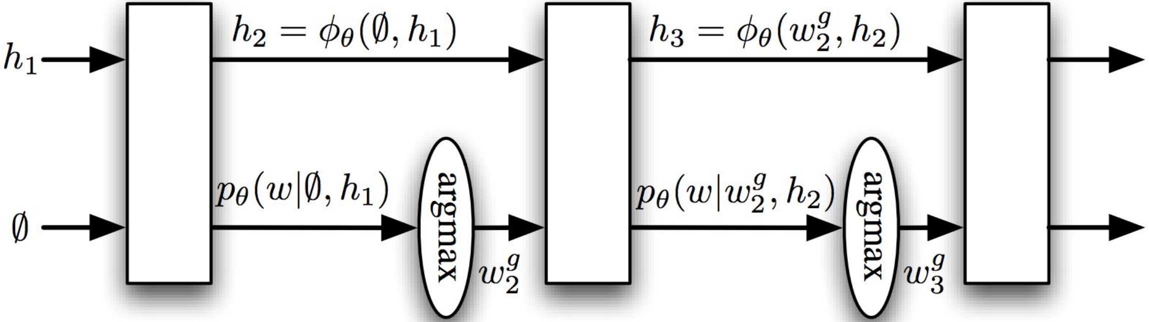
Testing

Model generates the entire sequence from scratch

Exposure Bias with Cross Entropy Training



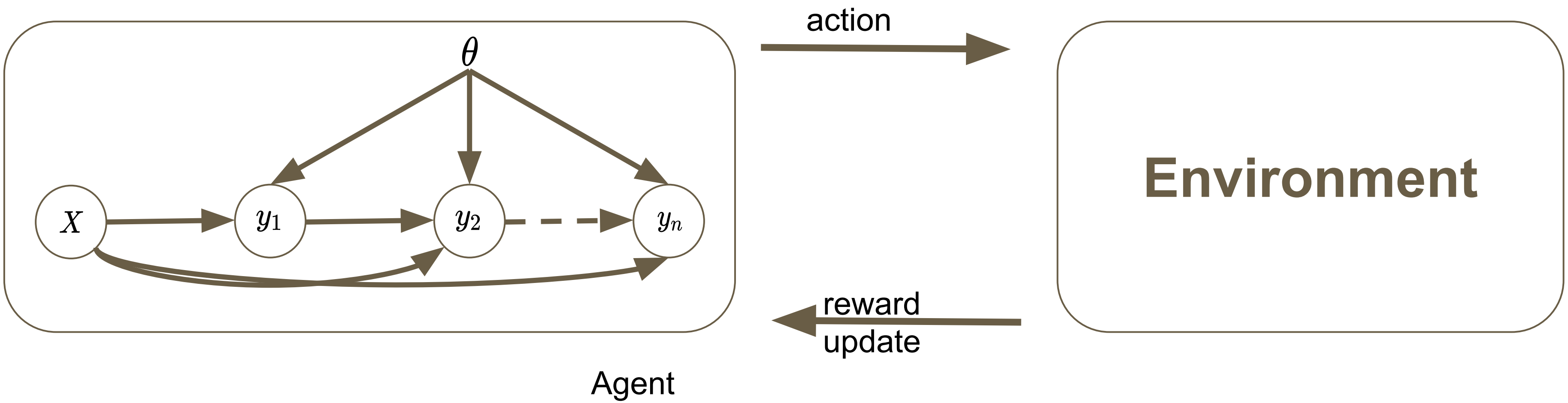
Training



Testing

(Ranzato et al., ICLR 2016)

Text Production as a Reinforcement Learning Problem



Policy Gradient to Optimize Task-Specific Metric

Goal of training is to find the parameters of the agent that **maximize the expected reward**

Loss is the **negative expected reward**

$$L(\theta) = -\mathbb{E}_{\hat{y} \sim p_\theta}[r(\hat{y})]$$

$$= -\sum_{\hat{y} \sim p_\theta} r(\hat{y}) p(\hat{y}|\theta)$$

$$\nabla L(\theta) = -\sum_{\hat{y} \sim p_\theta} r(\hat{y}) \nabla p(\hat{y}|\theta)$$

$$= -\mathbb{E}_{\hat{y} \sim p_\theta}[r(\hat{y}) \nabla \log p_\theta(\hat{y}|\theta)]$$

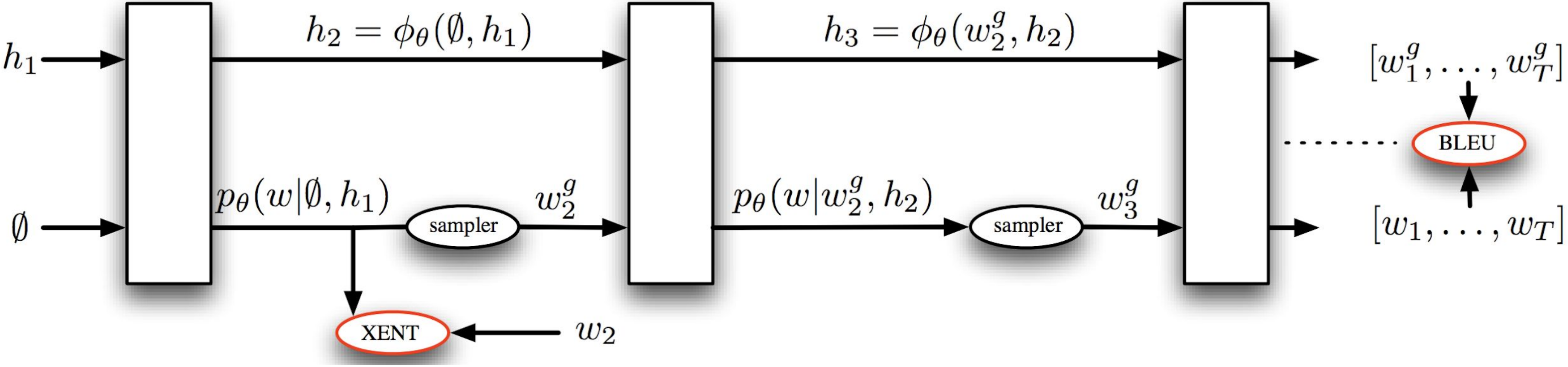
REINFORCE algorithm (Williams, 1992)

Policy Gradient to Optimize Task-Specific Metric

In practice, we approximate the **expected gradient using a single sample** for each training example

$$\nabla L(\theta) = -\mathbb{E}_{\hat{y} \sim p_\theta}[r(\hat{y}) \nabla \log p_\theta(\hat{y}|\theta)]$$
$$\approx -r(\hat{y}) \nabla \log p_\theta(\hat{y}|\theta)$$

Curriculum Learning



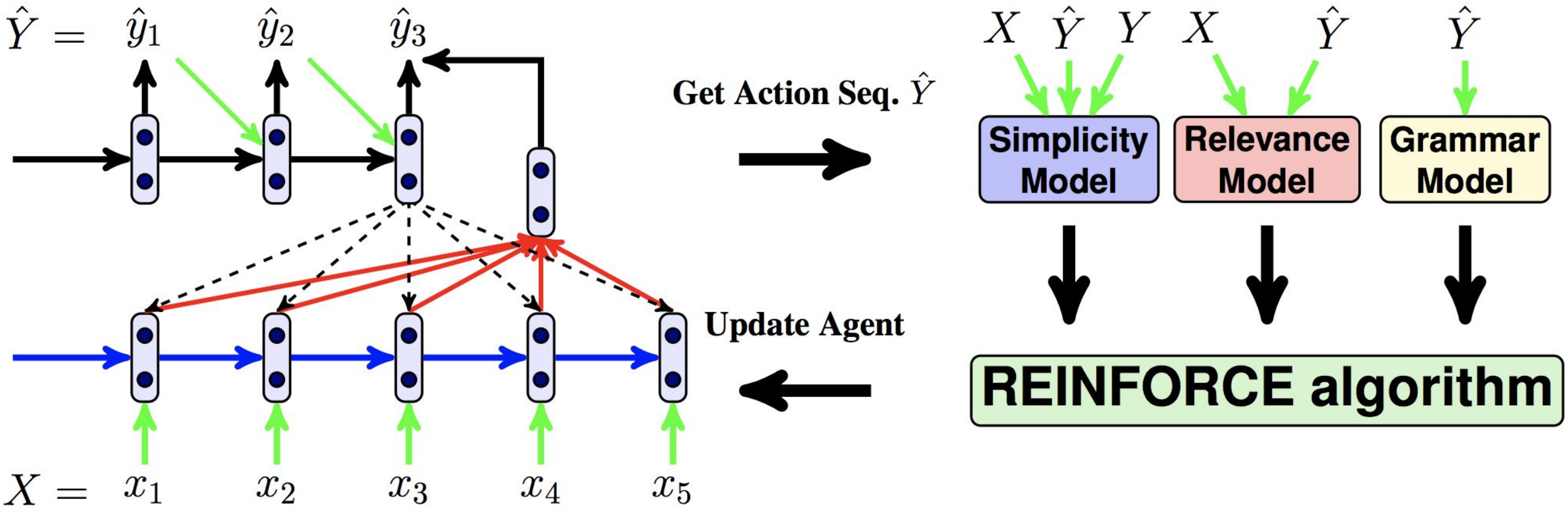
(Ranzato et al., ICLR 2016)

Curriculum Learning

Data: a set of sequences with their corresponding context.
Result: RNN optimized for generation.
Initialize RNN at random and set N^{XENT}, $N^{\text{XE+R}}$ and Δ;
for $s = T, 1, -\Delta$ **do**
    if $s == T$ **then**
        train RNN for N^{XENT} epochs using XENT only;
    else
        train RNN for $N^{\text{XE+R}}$ epochs. Use XENT loss in the first s steps, and REINFORCE (sampling from the model) in the remaining $T - s$ steps;
    end
end

(Ranzato et al., ICLR 2016)

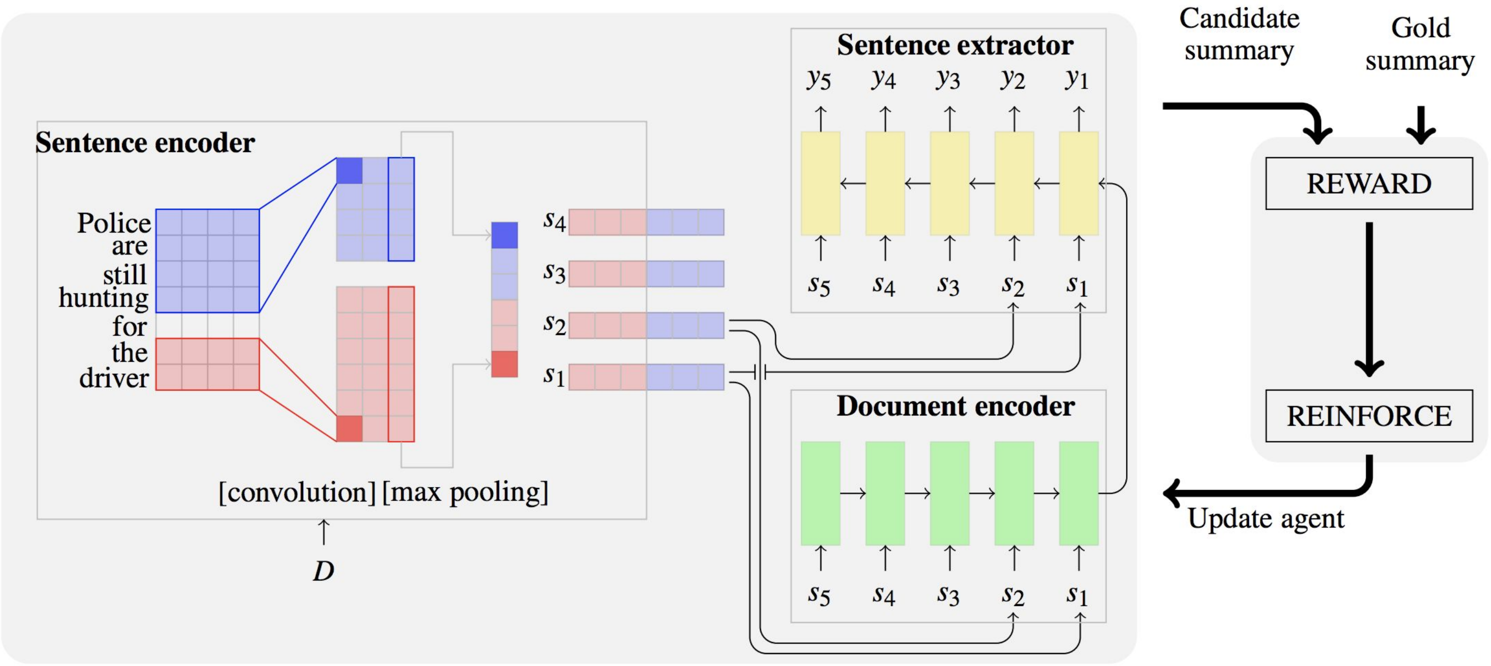
Sentence Simplification



Optimizes BLEU and SARI jointly

(Zhang and Lapata, 2017)

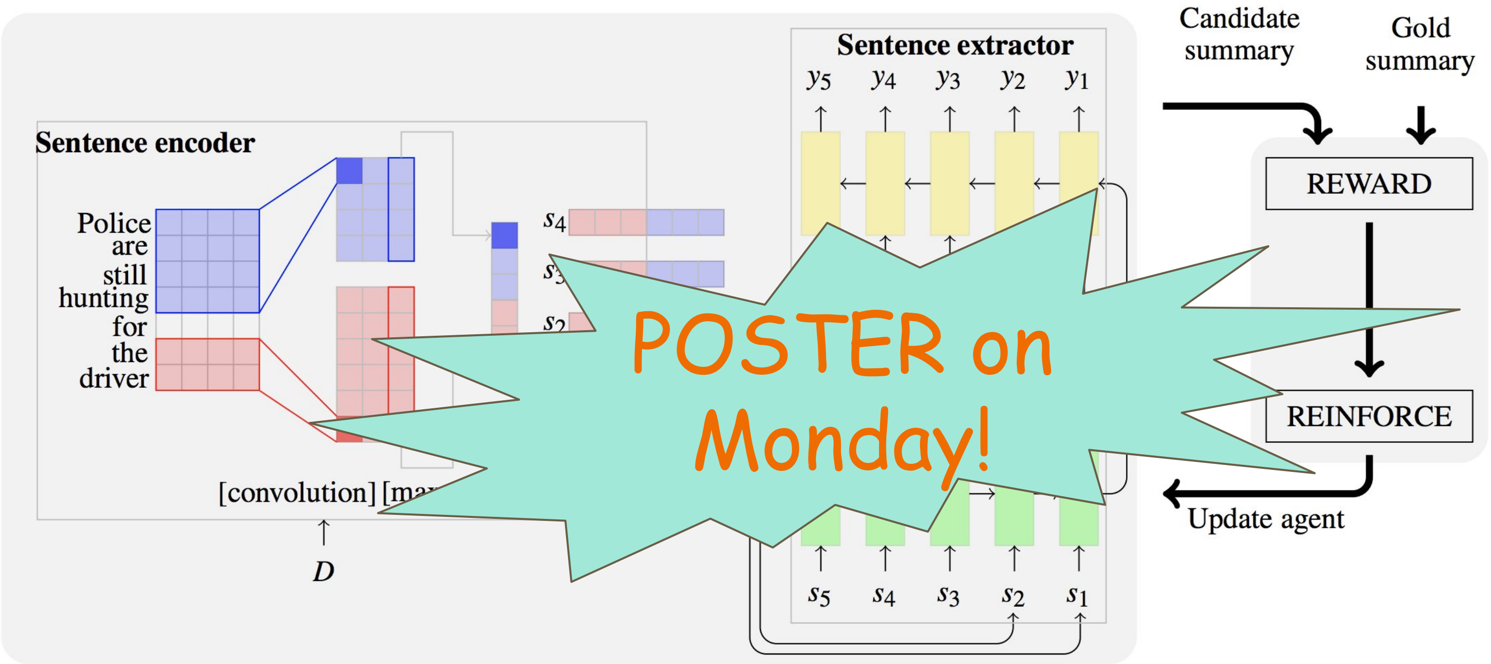
Extractive Document Summarization



Optimizes ROUGE scores

(Narayan et al, NAACL 2018)

Extractive Document Summarization



Optimizes ROUGE scores

(Narayan et al, NAACL 2018)

Abstractive Document Summarization

Optimizes ROUGE scores

- A Deep Reinforced Model for Abstractive Summarization (Paulus 2017)
- Multi-Reward Reinforced Summarization with Saliency and Entailment (Pasunuru and Bansal **NAACL** 2018)
- Deep Communicating Agents for Abstractive Summarization (Celikyilmaz et al **NAACL** 2018)

Abstractive Document Summarization

Optimizes ROUGE scores

- A Deep Reinforced Model for Abstrac… ation (P…ulus 2017)
- Multi-Reward Reinforced S… d Entailment (Pasunuru and Ba…
- Deep Communicating A… (Celikyilmaz et al **NAACL** 2018)

Last two papers are presented here!

Datasets, Challenges and Example Systems for Text Production

Datasets and Challenges for Neural NLG

DATA to text

WebNLG: Generating from RDF Data

E2E: Generating from Dialog Acts

Meaning Representations to text

SemEval Task 9: Generating from Abstract Meaning Representations

2018 Shared Task (SR'18)

Data Overview

	INPUT	Output	Domain	Lge	NLG
WebNLG	RDF	Text	15 domains	En	MicroPlanning
E2E	Dialog Act	Text	Restaurabt	En	MicroPlanning
SemEval	AMR	Stce	News	En	SR
SR	Dep. Trees	Stce	News	En, Fr, Sp	SR

Datasets and Challenges for T2T Generation

Simplification:

- Wikismall, Wikilarge, Newsela

Sentence Compression/Summarisation

- English Gigaword [Rush et al, 2015], DUC 2004 Test Set [Over et al., 2007], MSR-ATC Test Set [Toutanova et al. 2016], News Sentence-compression pairs [Filippova et al. 2015]

Paraphrasing

- PPDB (Ganitkevitch et al., 2013), Multiple-Translation Chinese (MTC) corpus (LDC)

Datasets and Challenges for Simplification

- WikiSmall (Zhu et al., 2010)
  - automatically-aligned complex-simple pairs from the ordinary-simple English Wikipedias.
  - Training: 89, 042 pairs. Test: 100
- WikiLarge (Zhang and Lapata, 2017)
  - Larger Wikipedia corpus aggregating pairs from Kauchak (2013), Woodsend and Lapata (2011), and WikiSmall.
  - All: 296,402 sentence pairs
- Newsela (Xu et al., 2015)
  - Training: 94208 sentence pairs, Test: 1076

Datasets for Paraphrasing

- ParaNMT (Wieting and Gimpel 2017)
  - back-translated paraphrase dataset
  - 50M+ back-translated paraphrases from the Czeng1.6 corpus
- PPDB (Ganitkevitch et al, 2013)
  - paraphrastic textual fragments extracted automatically from bilingual text
- Multiple-Translation Chinese (MTC) corpus (LDC)
  - 11 translations per input. Used for testing.

Datasets for Sentence Compression

English Gigaword [Rush et al, 2015]

- News: (First sentence, Headline). **Train**: 3.8M, **Test**: 189K

DUC 2004 [Over et al., 2007]

- (Sentence, Summary). **Test**: 500 input with 4 summaries each.

MSR-ATC [Toutanova et al. 2016]

- **Test**: 26K (Sentence, Summary)

News Sentence-compression pairs [Filippova et al. 2015]

- **Test**: 10K pairs

Datasets and Challenges for Summarisation

- CNN/DailyMail Story Highlights Generation dataset (Hermann et al. 2015)
- NY Times Summarization dataset (Sandhaus, 2008)

Datasets	# docs (train/val/test)	avg. document length		avg. summary length		vocabulary size	
		words	sentences	words	sentences	document	summary
CNN	90,266/1,220/1,093	760.50	33.98	45.70	3.59	343,516	89,051
DailyMail	196,961/12,148/10,397	653.33	29.33	54.65	3.86	563,663	179,966
NY Times	589,284/32,736/32,739	800.04	35.55	45.54	2.44	1,399,358	294,011

Multidocument summarisation DUC and TAC too small

Datasets and Challenges for Summarisation

- CNN/DailyMail Story Highlights Generation dataset (Hermann et al. 2015)
- NY Times Summarization dataset (Sandhaus, 2008)

Datasets	# docs (train/val/test)	avg. document length		avg. summary length		vocabulary size	
		words	sentences	words	sentences	document	summary
CNN	90,266/1,220/1,093	760.50	33.98	45.70	3.59	343,516	89,051
DailyMail	196,961/12,148/10,397	653.33	29.33	54.65	3.86	563,663	179,966
NY Times	589,284/32,736/32,739	800.04	35.55	45.54	2.44	1,399,358	294,011

Multidocument

Newsroom Dataset for Diverse Summarization

(Grusky et al. NAACL 2018)

Open Challenges

- Producing text in languages other than English
  - Multilingual SR Task, AMR-to-Chinese
  - Byte Pair Encoding (Sennrich et al 2016)
- Taking the Discourse Structure of input text into account
  - simplification, abstractive summarisation (Cohan et al 2018, Bosselut et al 2018)
- Structuring long output (Hierarchical Generation)
  - Story generation (Fan et al 2018), Poetry, Data to document
  - Transformer (Vaswani et al, 2017) and ConvSeq2Seq (Gehring et al, 2017) architecture
- Generating under constraints
  - length, emotion,style, user profile, syntax etc (Park et al, 2018)
  - VAE, Generative models

Thank you!

References

1992

- Simple statistical gradient-following algorithms for connectionist reinforcement learning, Ronald J. Williams, Machine Learning, 1992.

1994

- Learning long-term dependencies with gradient descent is difficult, Bengio Y., Simard P, Frasconi P., IEEE Trans Neural Netw. 1994. 5(2):157-66.

1997

- Long Short-Term Memory, Sepp Hochreiter and Jürgen Schmidhuber, Neural Computation 1997.

2001

- Automatic summarization, Indrajeet Mani, John Benjamins Publishing, 2001.

2002

- Multiple-Translation Chinese Corpus, LDC 2002T01, 2002.

2006

- Building Natural Language Generation Systems, Ehud Reiter and Robert Dale, Cambridge University Press, 2006.

References

2007

- DUC in context, Paul Over, Hoa Dang and Donna Harman, Information Processing and Management 2007.

2008

- The New York Times annotated corpus. Evan Sandhaus. Linguistic Data Consortium, Philadelphia, 6(12):e26752, 2008.

2010

- A Simple Domain-Independent Probabilistic Approach to Generation, Gabor Angeli, Percy Liang, Dan Klein, EMNLP 2010.

- A monolingual tree-based translation model for sentence simplification, Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. COLING 2010.

2011

- The First Surface Realisation Shared Task: Overview and Evaluation Results, Anja Belz, Michael White, Dominic Espinosa, Eric Kow, Deirdre Hogan, Amanda Stent. ENLG 2011.

References

2012

- Unsupervised concept-to-text generation with hypergraphs, Ioannis Konstas and Mirella Lapata, NAACL 2012.
- Concept-to-text generation via discriminative reranking, Ioannis Konstas and Mirella Lapata, ACL 2012.
- Structure-Driven Lexicalist Generation, Shashi Narayan, Claire Gardent, COLING 2012.

2013

- PPDB: The Paraphrase Database, Juri Ganitkevitch, Benjamin Van Durme and Chris Callison-Burch, NAACL 2013.

2014

- Hybrid Simplification using Deep Semantics and Machine Translation, Shashi Narayan and Claire Gardent, ACL 2014.
- Stochastic Language Generation in Dialogue Using Factored Language Models, Francois Mairesse, Steve Young, Computational Linguistics, 2014.

References

2014

- Sequence to Sequence Learning with Neural Networks, Ilya Sutskever, Oriol Vinyals, Quoc V. Le, arxiv:1409.3215, 2014.
- Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation, Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, Yoshua Bengio, EMNLP 2014.
- Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling, Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, Yoshua Bengio, NIPS - Deep Learning and Representation Learning Workshop 2014.
- Convolutional Neural Networks for Sentence Classification, Yoon Kim, EMNLP 2014.

2015

- Semantically Conditioned LSTM-based Natural Language Generation for Spoken Dialogue Systems, Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic,Pei-Hao Su, David Vandyke and Steve Young, 2015.
- Show and Tell: A Neural Image Caption Generator, Oriol Vinyals, Alexander Toshev, Samy Bengio, Dumitru Erhan, CVPR 2015.

References

2015

- Deep learning, Yann LeCun, Yoshua Bengio and Geoffrey Hinton, Nature 2015.
- Neural Machine Translation by Jointly Learning to Align and Translate, Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio, ICLR 2015.
- Sentence Compression by Deletion with LSTMs, Katja Filippova, Enrique Alfonseca, Carlos A. Colmenares, Lukasz Kaiser, Oriol Vinyals, EMNLP 2015.
- A Neural Attention Model for Abstractive Sentence Summarization, Alexander M. Rush, Sumit Chopra, Jason Weston, EMNLP 2015.
- Problems in Current Text Simplification Research: New Data Can Help, Wei Xu, Chris Callison-Burch, Courtney Napoles, TACL 2015.
- Teaching Machines to Read and Comprehend, Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, Phil Blunsom, NIPS 2015.

References

2016

- Paraphrase Generation from Latent-Variable PCFGs for Semantic Parsing, Shashi Narayan, Siva Reddy and Shay B. Cohen, INLG 2016.

- Unsupervised Sentence Simplification Using Deep Semantics, Shashi Narayan and Claire Gardent, INLG 2016.

- Deep Reinforcement Learning for Dialogue Generation, Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, Dan Jurafsky, EMNLP 2016.

- A Persona-Based Neural Conversation Model, Jiwei Li, Michel Galley, Chris Brockett, Georgios P. Spithourakis, Jianfeng Gao, Bill Dolan, ACL 2016.

- A Diversity-Promoting Objective Function for Neural Conversation Models, Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, Bill Dolan, NAACL 2016.

- Multi-domain Neural Network Language Generation for Spoken Dialogue Systems, Tsung-Hsien Wen, Milica Gasi c, Nikola Mrksic, Lina M. Rojas-Barahona, Pei-Hao Su, David Vandyke, Steve Young, 2016.

- Globally Coherent Text Generation with Neural Checklist Models, Chloe Kiddon, Luke Zettlemoyer and Yejin Choi, EMNLP 2016.

References

2016

- Incorporating Copying Mechanism in Sequence-to-Sequence Learning, Jiatao Gu, Zhengdong Lu, Hang Li, Victor O.K. Li, ACL 2016.
- Neural Summarization by Extracting Sentences and Words, Jianpeng Cheng, Mirella Lapata, ACL 2016.
- Pointing the Unknown Words, Caglar Gulcehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, Yoshua Bengio, ACL 2016.
- Modeling coverage for neural machine translation, Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li, ACL 2016.
- Neural Machine Translation of Rare Words with Subword Units, Rico Sennrich, Barry Haddow, Alexandra Birch, ACL 2016.
- Abstractive Text Summarization Using Sequence-to-Sequence RNNs and Beyond, Ramesh Nallapati, Bowen Zhou, Cicero Nogueira dos santos, Caglar Gulcehre, Bing Xiang, CoNLL 2016.
- Sequence Level Training with Recurrent Neural Networks, Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, Wojciech Zaremba, ICLR 2016.

References

2016

- A Dataset and Evaluation Metrics for Abstractive Compression of Sentences and Short Paragraphs, Kristina Toutanova, Chris Brockett, Ke M. Tran and Saleema Amershi, EMNLP 2016.

- Optimizing Statistical Machine Translation for Text Simplification, Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, Chris Callison-Burch, TACL 2016.

2017

- Creating Training Corpora for NLG Micro-Planning, Claire Gardent, Anastasia Shimorina, Shashi Narayan and Laura Perez-Beltrachini, ACL 2017.

- The WebNLG Challenge: Generating Text from RDF Data, Claire Gardent, Anastasia Shimorina, Shashi Narayan and Laura Perez-Beltrachini, INLG 2017.

- Split and Rephrase, Shashi Narayan, Claire Gardent, Shay B. Cohen and Anastasia Shimorina, EMNLP 2017.

- Sentence Simplification with Deep Reinforcement Learning, Xingxing Zhang, Mirella Lapata, EMNLP 2017.

- SemEval-2017 Task 9: Abstract Meaning Representation Parsing and Generation. http://alt.qcri.org/semeval2017/task9/

References

2017

- Shared Task Proposal: Multilingual Surface Realization Using Universal Dependency Trees, Simon Mille, Bernd Bohnet, Leo Wanner, Anja Belz, INLG 2017.

- Neural AMR: Sequence-to-Sequence Models for Parsing and Generation, Ioannis Konstas, Srinivasan Iyer, Mark Yatskar, Yejin Choi and Luke Zettlemoyer, ACL 2017.

- The E2E Dataset: New Challenges For End-to-End Generation, Jekaterina Novikova, Ondrej Dusek and Verena Rieser, SigDial Short 2017.

- Challenges in Data-to-Document Generation, Sam Wiseman, Stuart M. Shieber, Alexander M. Rush, EMNLP 2017.

- Get To The Point: Summarization with Pointer-Generator Networks, Abigail See, Peter J. Liu and Christopher D. Manning, ACL 2017.

- Abstractive Document Summarization with a Graph-Based Attentional Neural Model, Jiwei Tan and Xiaojun Wan, ACL 2017.

- Generating Natural Answers by Incorporating Copying and Retrieving Mechanisms in Sequence-to-Sequence Learning, Shizhu He, Cao Li, Kang Liu and Jun Zhao, ACL 2017.

References

2017

- A char-based seq2seq submission to the E2E NLG Challenge, Shubham Agarwal, Marc Dymetman and Eric Gaussier, E2E Challenge 2017.
- A surprisingly effective out-of-the-box char2char model on the e2e nlg challenge dataset, Shubham Agarwal and Marc Dymetman, SigDial 2017.
- Paraphrasing Revisited with Neural Machine Translation. Jonathan Mallinson; Rico Sennrich; Mirella Lapata, EACL 2017.
- Semi-Supervised Classification with Graph Convolutional Networks, Thomas N. Kipf, Max Welling, ICLR 2017.
- Encoding Sentences with Graph Convolutional Networks for Semantic Role Labeling, Diego Marcheggiani, Ivan Titov. EMNLP, 2017.
- Graph Convolutional Encoders for Syntax-aware Neural Machine Translation, Joost Bastings, Ivan Titov, Wilker Aziz, Diego Marcheggiani, Khalil Sima'an. EMNLP, 2017.
- Tutorial: Neural Methods for Semantic Role Labeling, Diego Marcheggiani, Michael Roth, Ivan Titov, Benjamin Van Durme, EMNLP 2017.

References

2017

- Selective Encoding for Abstractive Sentence Summarization, Qingyu Zhou, Nan Yang, Furu Wei, Ming Zhou, ACL 2017.
- A Deep Reinforced Model for Abstractive Summarization, Romain Paulus, Caiming Xiong, Richard Socher, abs/1705.04304, 2017.
- Attention Is All You Need, Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, abs/1706.03762, 2017.
- Convolutional Sequence to Sequence Learning, Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, Yann N. Dauphin, abs/1705.03122, 2017.
- ParaNMT-50M: Pushing the Limits of Paraphrastic Sentence Embeddings with Millions of Machine Translations, John Wieting, Kevin Gimpel, abs/1711.05732, 2017.

References

2018

- A Graph-to-Sequence Model for AMR-to-Text Generation. Linfeng Song, Yue Zhang, Zhiguo Wang and Daniel Gildea, ACL 2018.
- ACL 2018 Workshop on Multilingual Surface Realization. http://taln.upf.edu/pages/msr2018-ws/
- GTR-LSTM: A Triple Encoder for Sentence Generation from RDF Data. Bayu Distiawan Trisedya, Jianzhong Qi, Rui Zhang and Wei Wang, ACL 2018.
- Bootstrapping Generators from Noisy Data, Laura Perez-Beltrachini, Mirella Lapata, NAACL 2018.
- DEEP COMMUNICATING AGENTS FOR ABSTRACTIVE SUMMARIZATION, Asli Celikyilmaz, Antoine Bosselut, Xiaodong He and Yejin Choi, NAACL 2018.
- NEWSROOM: A DATASET OF 1.3 MILLION SUMMARIES WITH DIVERSE EXTRACTIVE STRATEGIES, Max Grusky, Mor Naaman and Yoav Artzi, NAACL 2018.
- RANKING SENTENCES FOR EXTRACTIVE SUMMARIZATION WITH REINFORCEMENT LEARNING, Shashi Narayan, Shay B. Cohen and Mirella Lapata, NAACL 2018.

References

2018

- MULTI-REWARD REINFORCED SUMMARIZATION WITH SALIENCY AND ENTAILMENT, Ramakanth Pasunuru and Mohit Bansal, NAACL 2018.
- Document Modeling with External Attention for Sentence Extraction, Shashi Narayan, Ronald Cardenas, Nikos Papasarantopoulos, Shay B. Cohen, Mirella Lapata, Jiangsheng Yu and Yi Chang, ACL 2018.
- Generating syntactically varied realisations from AMR graphs, Kris Cao, Stephen Clark, abs/1804.07707, 2018.
- Bootstrapping Generators from Noisy Data, Laura Perez-Beltrachini, Mirella Lapata, NAACL 2018.
- Hierarchical Neural Story Generation. Angela Fan, Mike Lewis and Yann Dauphin. ACL 2018.
- A Hierarchical Latent Structure for Variational Conversation Modeling, Yookoon Park, Jaemin Cho and Gunhee Kim, NAACL 2018.
- A Discourse-Aware Attention Model for Abstractive Summarization of Long Documents, Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, Nazli Goharian, NAACL 2018.

References

2018

- Discourse-Aware Neural Rewards for Coherent Text Generation, Antoine Bosselut, Asli Celikyilmaz, Xiaodong He, Jianfeng Gao, Po-Sen Huang, Yejin Choi, NAACL 2018.

Others:

- Understanding LSTM Networks: http://colah.github.io/posts/2015-08-Understanding-LSTMs/