Natural Language Processing (CS-472) Spring-2023

Muhammad Naseer Bajwa
Assistant Professor,
Department of Computing, SEECS
Co-Principal Investigator,
Deep Learning Lab, NCAI
NUST, Islamabad
naseer.bajwa@seecs.edu.pk

Overview of this week’s lecture

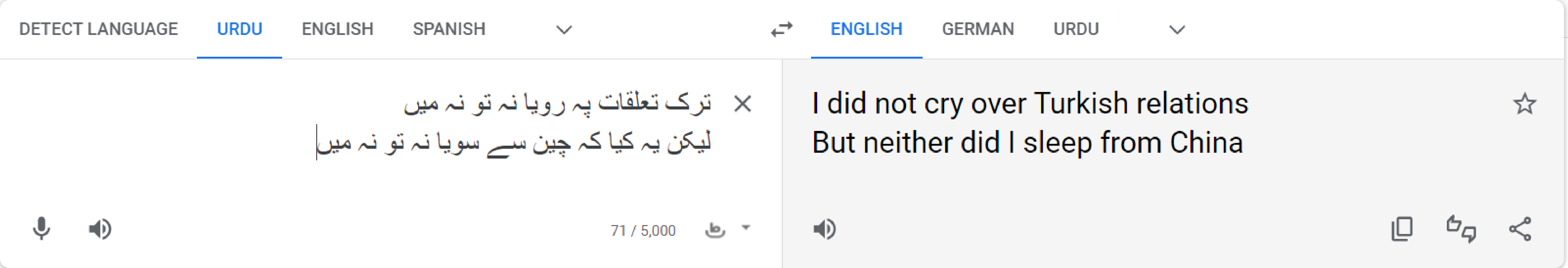
Natural Language Generation (NLG)

- Recap of NLG studied so far
- Revision and extension of decoding algorithms
- Suitable approaches for various NLG tasks
- Evaluation of NLG

Natural Language Generation generates coherent and useful natural language

Natural Language Generation generates coherent and useful natural language

- The NLG is commonly a vital component of many other NLP tasks.
    - Machine Translation



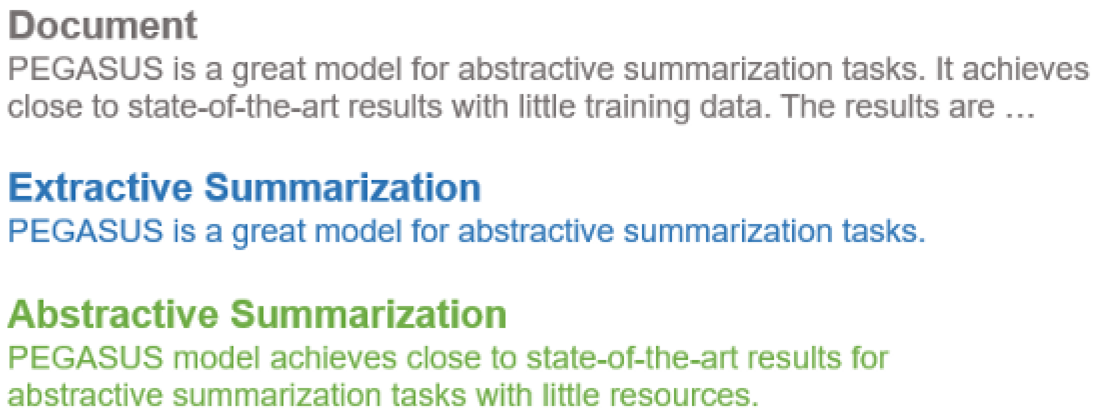
ترک تعلقات پہ رویا نہ تو نہ میں
لیکن یہ کیا کہ چین سے سویا نہ تو نہ میں

I did not cry over Turkish relations
But neither did I sleep from China

Natural Language Generation generates coherent and useful natural language

- The NLG is commonly a vital component of many other NLP tasks.

  - Machine Translation
  - Summarisation



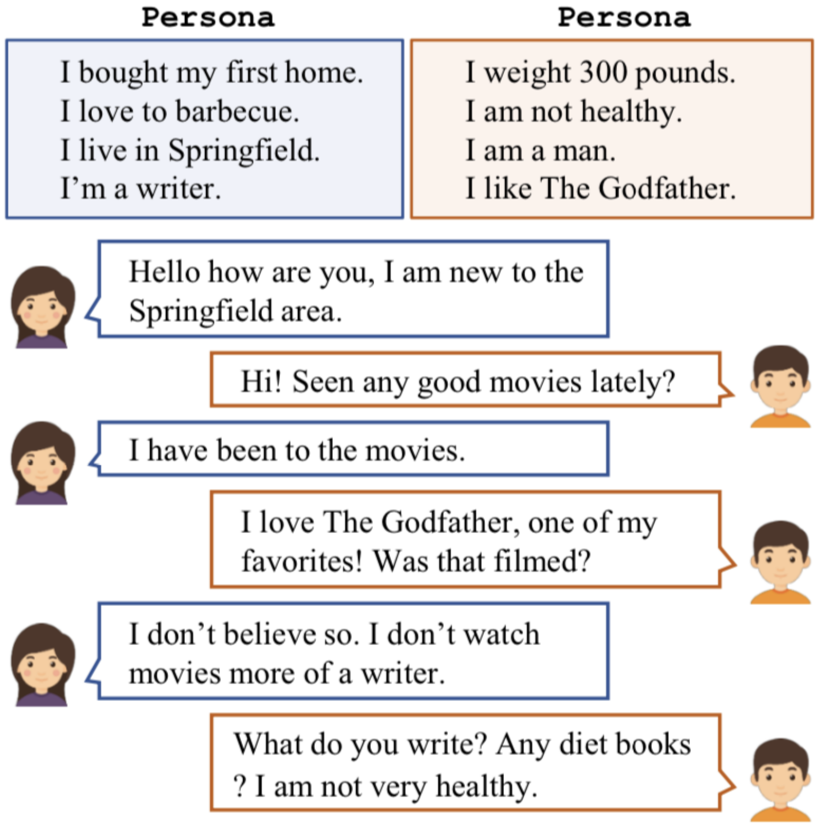
Document
PEGASUS is a great model for abstractive summarization tasks. It achieves close to state-of-the-art results with little training data. The results are …

Extractive Summarization
PEGASUS is a great model for abstractive summarization tasks.

Abstractive Summarization
PEGASUS model achieves close to state-of-the-art results for abstractive summarization tasks with little resources.

Natural Language Generation generates coherent and useful natural language

- The NLG is commonly a vital component of many other NLP tasks.
  - Machine Translation
  - Summarisation
  - Dialogue Generation



Persona

I bought my first home.
I love to barbecue.
I live in Springfield.
I'm a writer.

Persona

I weight 300 pounds.
I am not healthy.
I am a man.
I like The Godfather.

Hello how are you, I am new to the Springfield area.

Hi! Seen any good movies lately?

I have been to the movies.

I love The Godfather, one of my favorites! Was that filmed?

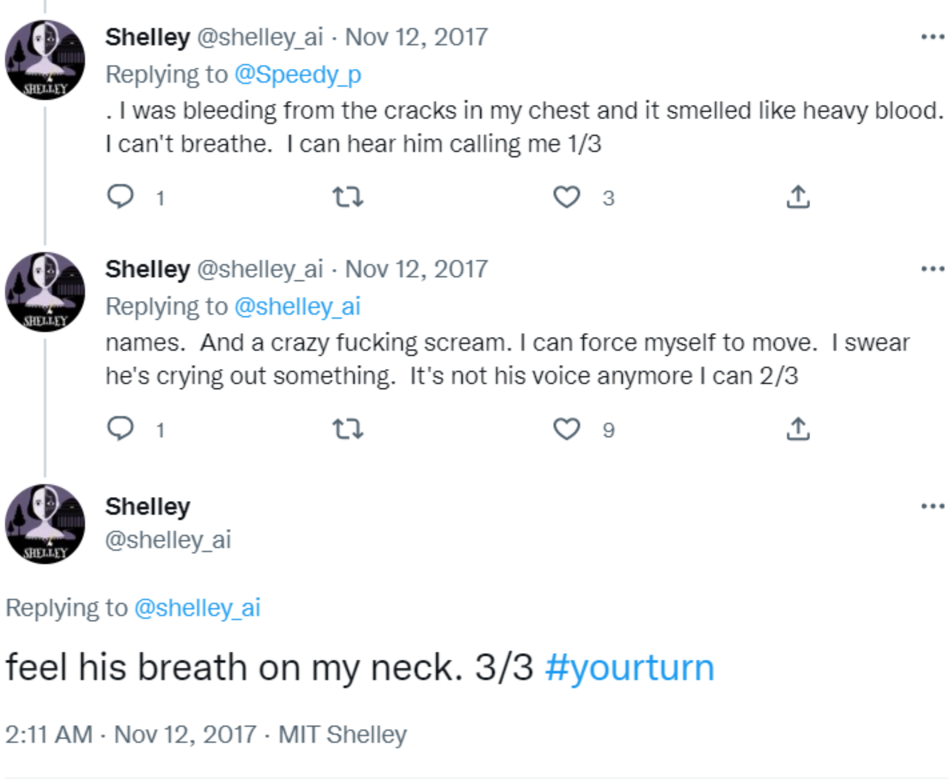
I don't believe so. I don't watch movies more of a writer.

What do you write? Any diet books ? I am not very healthy.

A clippled dialogue from PERSONA-CHAT.

Natural Language Generation generates coherent and useful natural language

- The NLG is commonly a vital component of many other NLP tasks.
  - Machine Translation
  - Summarisation
  - Dialogue Generation
  - Creative Writing



Shelley @shelley_ai · Nov 12, 2017
Replying to @Speedy_p
. I was bleeding from the cracks in my chest and it smelled like heavy blood. I can't breathe. I can hear him calling me 1/3
1 3

Shelley @shelley_ai · Nov 12, 2017
Replying to @shelley_ai
names. And a crazy fucking scream. I can force myself to move. I swear he's crying out something. It's not his voice anymore I can 2/3
1 9

Shelley
@shelley_ai

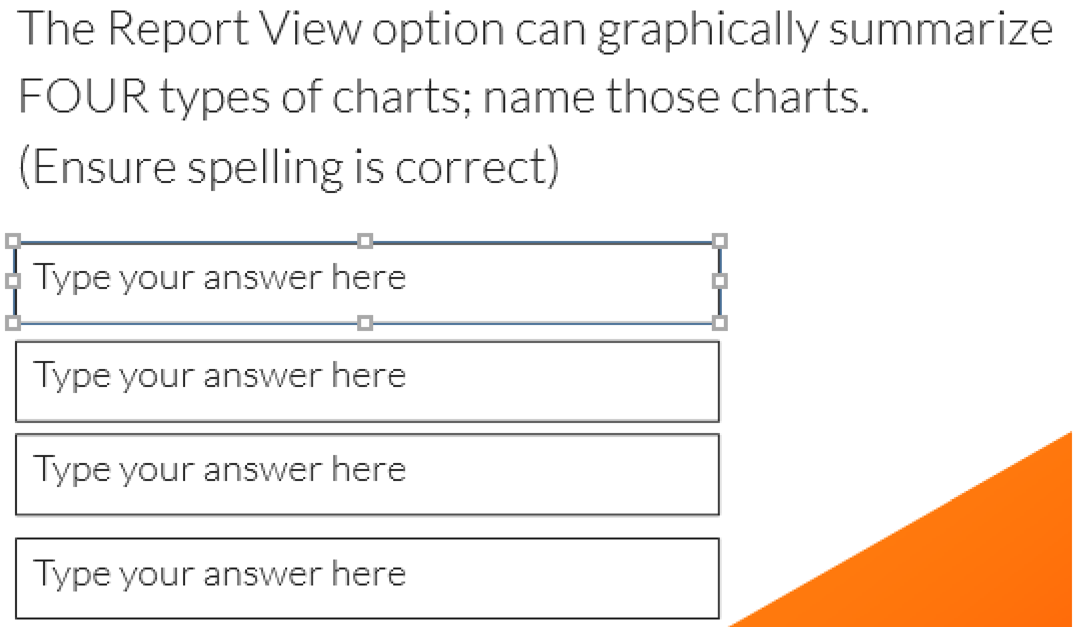
Replying to @shelley_ai

feel his breath on my neck. 3/3 #yourturn

2:11 AM · Nov 12, 2017 · MIT Shelley

Natural Language Generation generates coherent and useful natural language

- The NLG is commonly a vital component of many other NLP tasks.
  - Machine Translation
  - Summarisation
  - Dialogue Generation
  - Creative Writing
  - Freeform Question Answering



The Report View option can graphically summarize FOUR types of charts; name those charts. (Ensure spelling is correct)

Type your answer here

Type your answer here

Type your answer here

Type your answer here

Natural Language Generation generates coherent and useful natural language

- The NLG is commonly a vital component of many other NLP tasks.
  - Machine Translation
  - Summarisation
  - Dialogue Generation
  - Creative Writing
  - Freeform Question Answering
  - Image Captioning

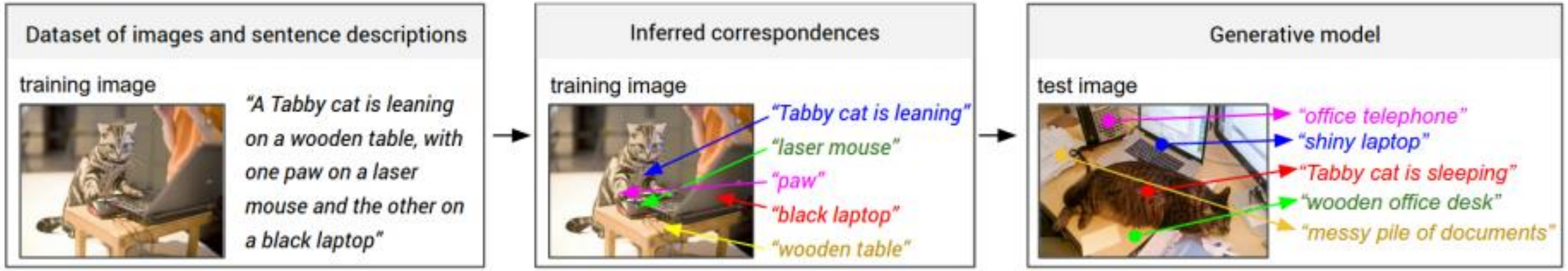


Figure 2. Overview of our approach. A dataset of images and their sentence descriptions is the input to our model (left). Our model first infers the correspondences (middle, Section 3.1) and then learns to generate novel descriptions (right, Section 3.2).

Natural Language Generation generates coherent and useful natural language

- The NLG is commonly a vital component of many other NLP tasks.
  - Machine Translation
  - Summarisation
  - Dialogue Generation
  - Creative Writing
  - Freeform Question Answering
  - Image Captioning
  - Data to Text Generation

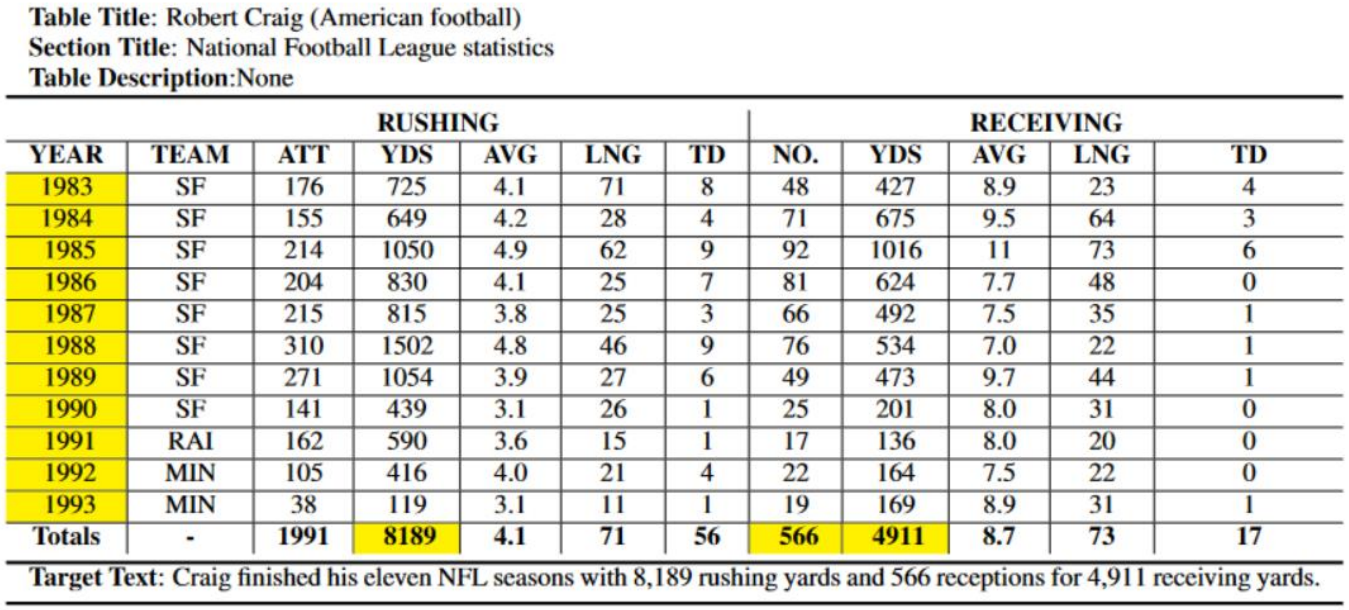


Table Title: Robert Craig (American football)
Section Title: National Football League statistics
Table Description:None

		RUSHING					RECEIVING				
YEAR	TEAM	ATT	YDS	AVG	LNG	TD	NO.	YDS	AVG	LNG	TD
1983	SF	176	725	4.1	71	8	48	427	8.9	23	4
1984	SF	155	649	4.2	28	4	71	675	9.5	64	3
1985	SF	214	1050	4.9	62	9	92	1016	11	73	6
1986	SF	204	830	4.1	25	7	81	624	7.7	48	0
1987	SF	215	815	3.8	25	3	66	492	7.5	35	1
1988	SF	310	1502	4.8	46	9	76	534	7.0	22	1
1989	SF	271	1054	3.9	27	6	49	473	9.7	44	1
1990	SF	141	439	3.1	26	1	25	201	8.0	31	0
1991	RAI	162	590	3.6	15	1	17	136	8.0	20	0
1992	MIN	105	416	4.0	21	4	22	164	7.5	22	0
1993	MIN	38	119	3.1	11	1	19	169	8.9	31	1
Totals	**-**	**1991**	**8189**	**4.1**	**71**	**56**	**566**	**4911**	**8.7**	**73**	**17**

Target Text: Craig finished his eleven NFL seasons with 8,189 rushing yards and 566 receptions for 4,911 receiving yards.

Table 4: An example in the ToTTo dataset that involves numerical reasoning over the table structure.

Huang, Zhiheng, et al. "Improve transformer models with better relative position embeddings." arXiv preprint arXiv:2009.13658 (2020).

Let’s recap NLG discussed so far in the course

- Language Modelling is the task of predicting the next word in a sequence given the words generated so far.

$$P(y^{<t>} | y^{<1>}, y^{<2>}, \dots, y^{<t-1>})$$

  - It is a model that learns conditional probabilities.
  - It can be an RNN-based LM or Attention-based LM.
  - Not much useful on its own.

Let’s recap NLG discussed so far in the course

- Language Modelling is the task of predicting the next word in a sequence given the words generated so far.

$$P(y^{<t>}|y^{<1>}, y^{<2>}, \ldots, y^{<t-1>})$$

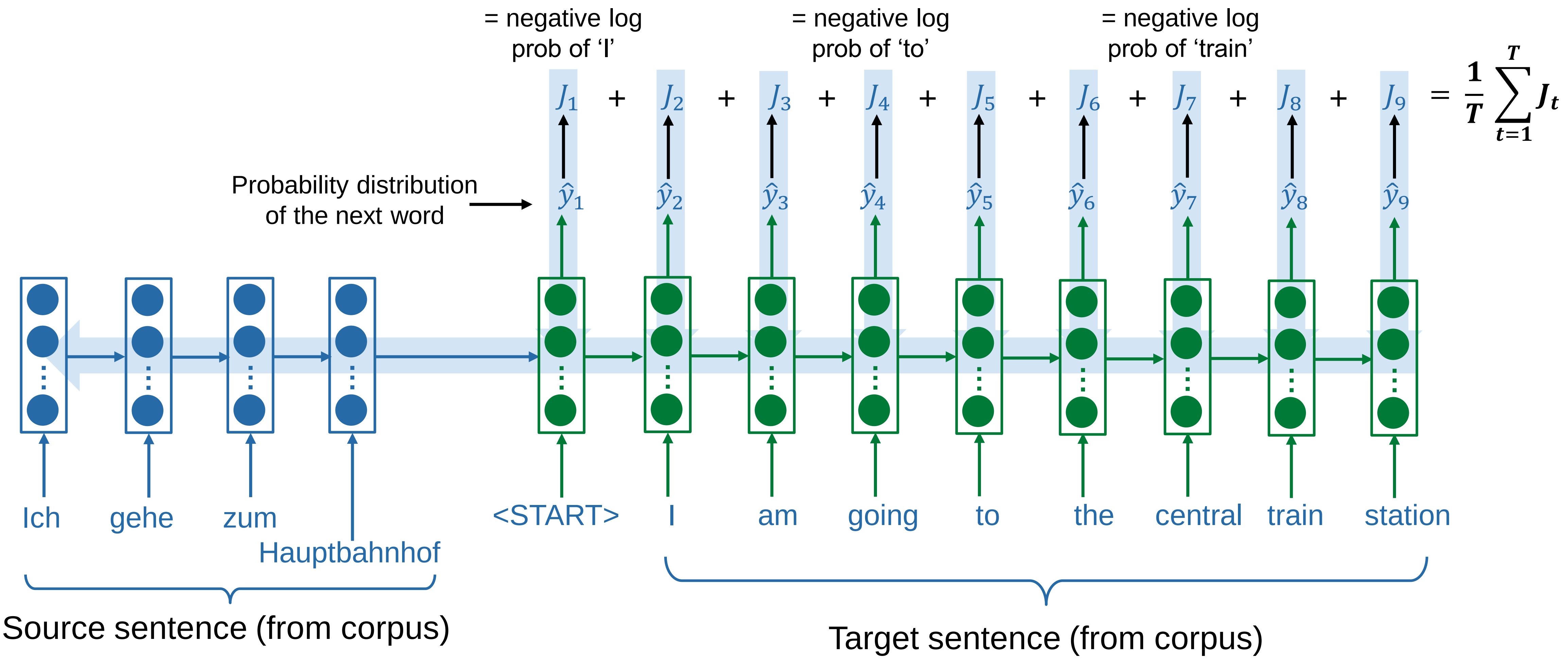
  - It is a model that learns conditional probabilities.
  - It can be an RNN-based LM or Attention-based LM.
  - Not much useful on its own.

- A conditional Language Model uses another input in addition to previously generated words to produce the next word.

$$P(y^{<t>}|y^{<1>}, y^{<2>}, \ldots, y^{<t-1>}, x)$$

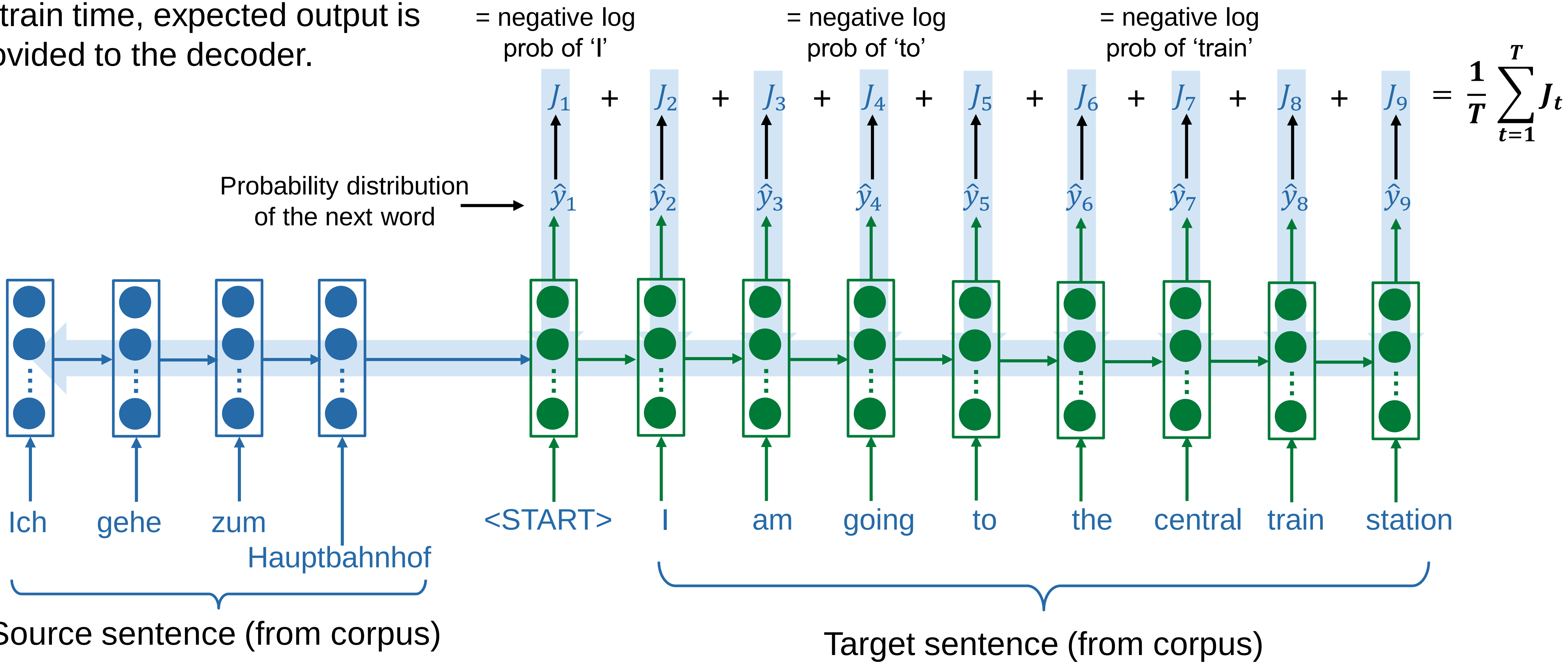
  - Machine Translation (x: source sentence, y: target sentence)
  - Summarisation (x: source document, y: summarised document)
  - Dialogue Generation (x: dialogue history, y: next response)

Language Models are trained using Teacher Forcing


= negative log prob of 'I'
= negative log prob of 'to'
= negative log prob of 'train'
$J_1 + J_2 + J_3 + J_4 + J_5 + J_6 + J_7 + J_8 + J_9 = \frac{1}{T}\sum_{t=1}^{T} J_t$
Probability distribution of the next word
$\hat{y}_1$ $\hat{y}_2$ $\hat{y}_3$ $\hat{y}_4$ $\hat{y}_5$ $\hat{y}_6$ $\hat{y}_7$ $\hat{y}_8$ $\hat{y}_9$
Ich
gehe
zum
Hauptbahnhof
<START>
I
am
going
to
the
central
train
station
Source sentence (from corpus)
Target sentence (from corpus)

Language Models are trained using Teacher Forcing

- At train time, expected output is provided to the decoder.



= negative log prob of ‘I’
= negative log prob of ‘to’
= negative log prob of ‘train’
$J_1 + J_2 + J_3 + J_4 + J_5 + J_6 + J_7 + J_8 + J_9 = \frac{1}{T}\sum_{t=1}^{T} J_t$
Probability distribution of the next word
$\hat{y}_1$ $\hat{y}_2$ $\hat{y}_3$ $\hat{y}_4$ $\hat{y}_5$ $\hat{y}_6$ $\hat{y}_7$ $\hat{y}_8$ $\hat{y}_9$
Ich
gehe
zum
Hauptbahnhof
<START>
I
am
going
to
the
central
train
station
Source sentence (from corpus)
Target sentence (from corpus)

What happens at each time step of training and inference?

- At each time step during training, the model computes a vector of scores for each token in the vocabulary, $s \in \mathbb{R}^V$.

$$s = f(\{x_{<t}\}, \theta)$$

$f(.)$ is the model

What happens at each time step of training and inference?

- At each time step during training, the model computes a vector of scores for each token in the vocabulary, $s \in \mathbb{R}^V$.

$$s = f(\{x_{<t}\}, \theta)$$

$f(.)$ is the model

- A probability distribution P is calculated over $w \in V$ using these scores.

$$P(y'_t = w | \{x_{<t}\}) = \frac{\exp(s_w)}{\sum_{w' \in V} \exp(s_{w'})}$$

What happens at each time step of training and inference?

- At each time step during training, the model computes a vector of scores for each token in the vocabulary, $s \in \mathbb{R}^V$.

$$s = f(\{x_{<t}\}, \theta)$$

$f(.)$ is the model

- A probability distribution P is calculated over $w \in V$ using these scores.

$$P(y'_t = w | \{x_{<t}\}) = \frac{\exp(s_w)}{\sum_{w' \in V} \exp(s_{w'})}$$

- At each time step during inference, the decoding algorithm defines a function to select a token from this distribution.

$$\hat{y}_t = g(P(y'_t | \{x_{<t}\}))$$

$g(.)$ is the decoding algorithm

What happens at each time step of training and inference?

- At each time step during training, the model computes a vector of scores for each token in the vocabulary, $s \in \mathbb{R}^V$.

$$s = f(\{x_{<t}\}, \theta)$$

$f(.)$ is the model

- A probability distribution P is calculated over $w \in V$ using these scores.

$$P(y'_t = w | \{x_{<t}\}) = \frac{\exp(s_w)}{\sum_{w' \in V} \exp(s_{w'})}$$

- At each time step during inference, the decoding algorithm defines a function to select a token from this distribution.

$$\hat{y}_t = g(P(y'_t | \{x_{<t}\}))$$

$g(.)$ is the decoding algorithm

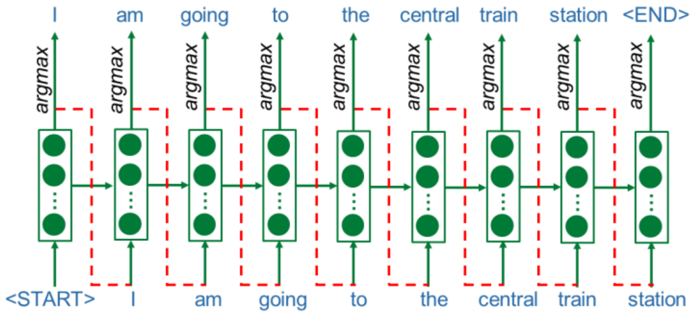
- The model is trained to minimise the negative log likelihood of predicting the next token in the sequence.

$$\mathcal{L}_t = -\log P(y_t | \{y_{<t}\})$$

Sum $\mathcal{L}_t$ for the entire sequence

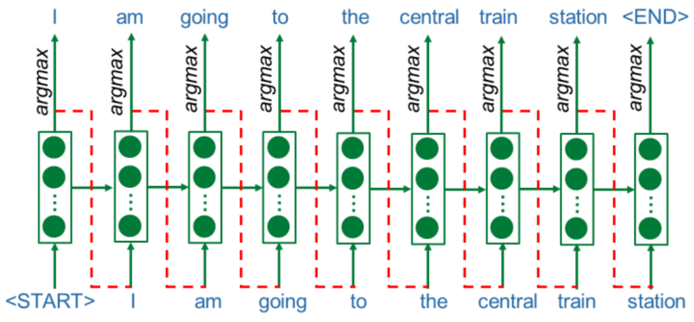
Decoding algorithms formulate the text using the words generated by LM

- Greedy Decoding:
  - At every decoder time step, select only the most probable next word.
  - Use this word as the autoregressive input at the next time step.
  - Stop when <END> is generated or MAX_LENGTH is reached.
  - Output could be ungrammatical, unnatural, and/or nonsensical due to lack of backtracking.

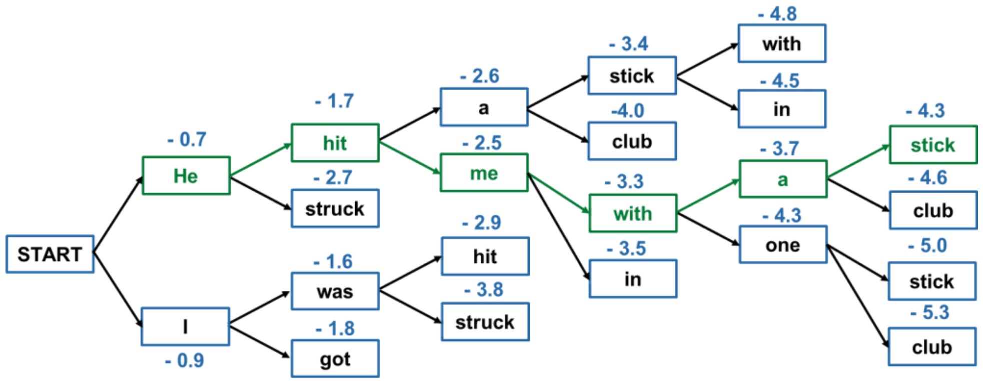


Decoding algorithms formulate the text using the words generated by LM

- Greedy Decoding:
  - At every decoder time step, select only the most probable next word.
  - Use this word as the autoregressive input at the next time step.
  - Stop when <END> is generated or MAX_LENGTH is reached.
  - Output could be ungrammatical, unnatural, and/or nonsensical due to lack of backtracking.



- Beam Search Decoding:
  - Find the most probable sequence by tracking multiple hypotheses simultaneously.
  - At every time step, track k most probable hypotheses.
  - Stop when stopping criteria are fulfilled.
  - Pick the sequence with the highest probability (adjusting for sequence length).



How the output changes with varying beam size?

- Beam search with smaller k will tend to behave more like greedy decoding.
    - The sentence could be grammatically incorrect, unnatural and nonsense.

Tu, Zhaopeng, et al. "Neural machine translation with reconstruction." Thirty-first AAAI conference on artificial intelligence. 2017.

Koehn, Philipp, and Rebecca Knowles. "Six challenges for neural machine translation." arXiv preprint arXiv:1706.03872 (2017).

How the output changes with varying beam size?

- Beam search with smaller k will tend to behave more like greedy decoding.
  - The sentence could be grammatically incorrect, unnatural and nonsense.
- Beam search with larger k will tend to behave more like exhaustive search.
  - No grammatically incorrect or disjointed output.
  - Increasing k too much may actually reduce BLEU score.
    - Because decoder starts to produce smaller sequences.
  - In open-ended tasks, too large k results in more generic output.

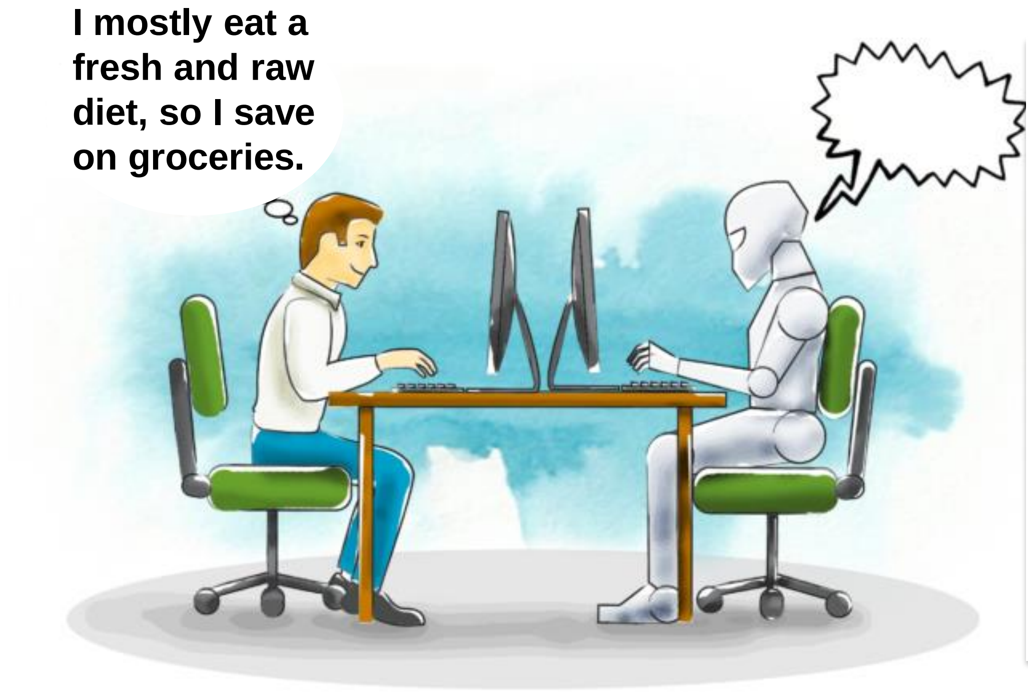
Tu, Zhaopeng, et al. "Neural machine translation with reconstruction." Thirty-first AAAI conference on artificial intelligence. 2017.
Koehn, Philipp, and Rebecca Knowles. "Six challenges for neural machine translation." arXiv preprint arXiv:1706.03872 (2017).

How the output changes with varying beam size?

- Beam search with smaller k will tend to behave more like greedy decoding.
  - The sentence could be grammatically incorrect, unnatural and nonsense.
- Beam search with larger k will tend to behave more like exhaustive search.
  - No grammatically incorrect or disjointed output.
  - Increasing k too much may actually reduce BLEU score.
    - Because decoder starts to produce smaller sequences.
  - In open-ended tasks, too large k results in more generic output.
- An optimal solution with respect to probability scores does not necessarily give high BLEU score.

Tu, Zhaopeng, et al. "Neural machine translation with reconstruction." Thirty-first AAAI conference on artificial intelligence. 2017.
Koehn, Philipp, and Rebecca Knowles. "Six challenges for neural machine translation." arXiv preprint arXiv:1706.03872 (2017).

Larger beam size may result in generic responses of chat bots

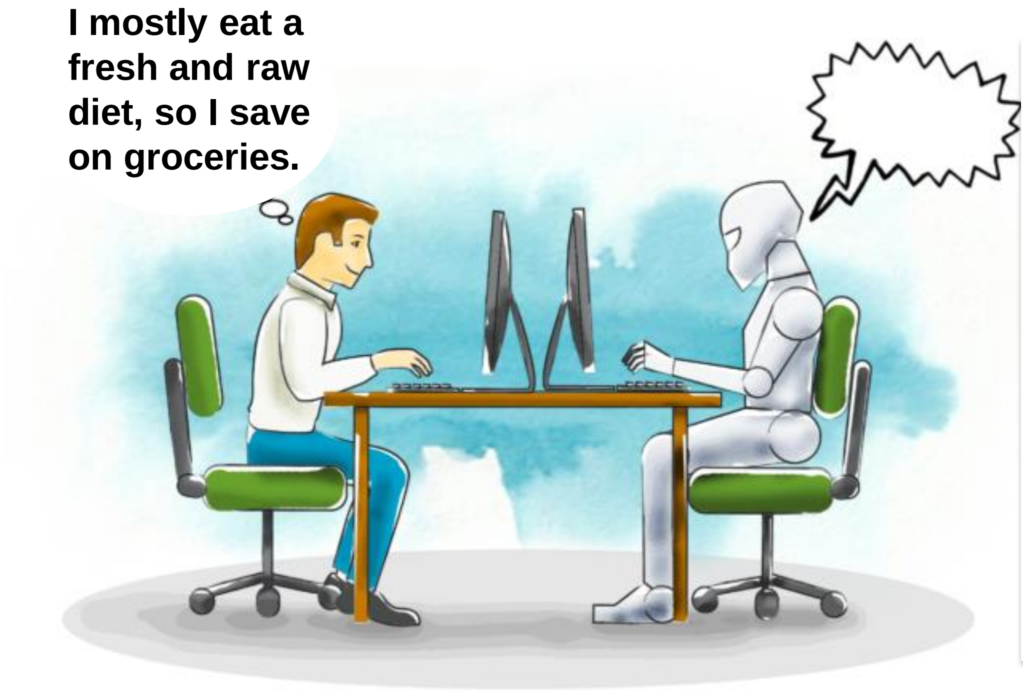


Model response	Beam Size
I love to eat healthy and eat healthy	1
That is a good thing to have	2
I am a nurse so I do not eat raw food	3
I am a nurse so I am a nurse	4
Do you have any hobbies?	5
What do you do for a living?	6
What do you do for a living?	7

Responses generated using PersonaChat dataset

https://www.kaggle.com/datasets/atharvjairath/personachat

Larger beam size may result in generic responses of chat bots



Model response	Beam Size
I love to eat healthy and eat healthy	1
That is a good thing to have	2
I am a nurse so I do not eat raw food	3
I am a nurse so I am a nurse	4
Do you have any hobbies?	5
What do you do for a living?	6
What do you do for a living?	7

Responses generated using PersonaChat dataset

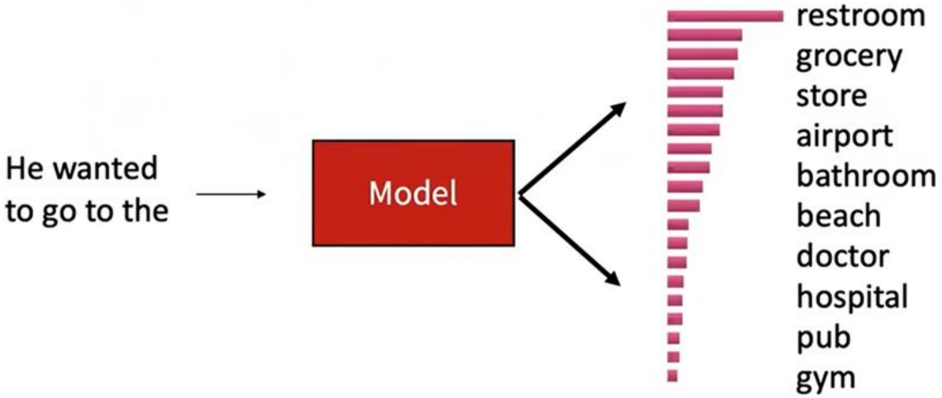
Smaller k:
On topic
Less fluent

Larger k:
Off topic
More fluent
Converges to safe correct response

https://www.kaggle.com/datasets/atharvjairath/personachat

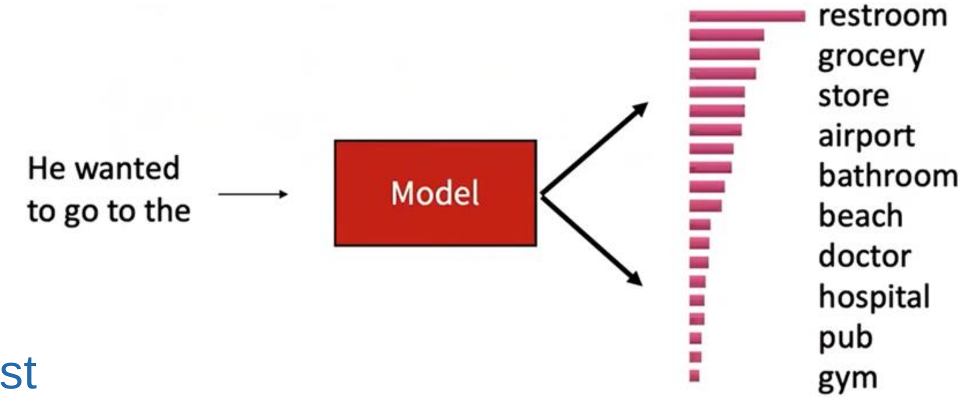
Sampling based decoding is more efficient method of selecting next words

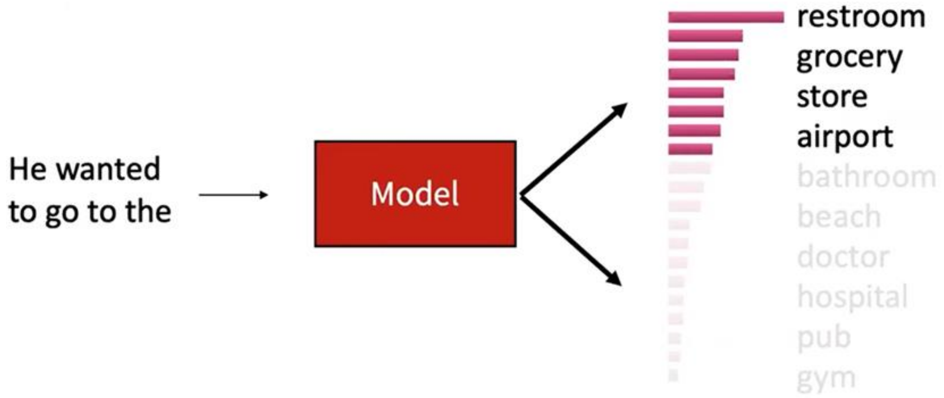
- Simple random sampling
  - At every decoder time step t, pick one word randomly from the probability distribution $P^{<t>}$.



Sampling based decoding is more efficient method of selecting next words

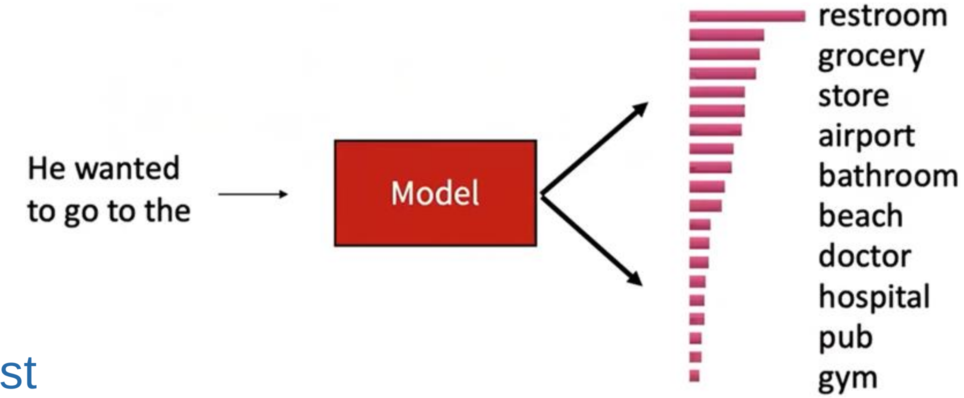
- Simple random sampling
  - At every decoder time step t, pick one word randomly from the probability distribution $P^{<t>}$.
- Top-N sampling
  - At every decoder time step t, pick a word randomly from N most probable words in probability distribution $P^{<t>}$.

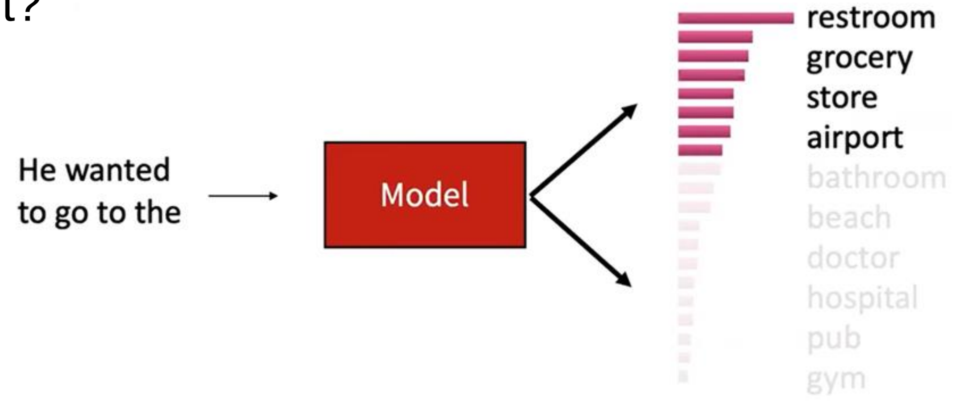




Sampling based decoding is more efficient method of selecting next words

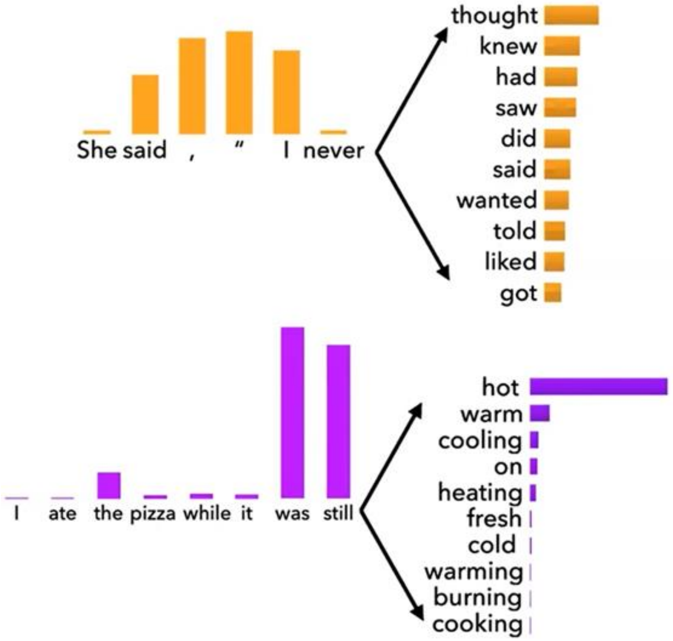
- Simple random sampling
    - At every decoder time step t, pick one word randomly from the probability distribution $P^{<t>}$.
- Top-N sampling
    - At every decoder time step t, pick a word randomly from N most probable words in probability distribution $P^{<t>}$.
- What would be the effect of varying N from 1 to V on decoder output?
    - Increasing N gives more diverse / risky outputs.
    - Decreasing N gives more specific / safe outputs.





Issues with top-N sampling

- The probability distributions to sample from may be dynamic.
  - When $P^{<t>}$ is flatter, a limited N removes many viable options.
  - When $P^{<t>}$ is peakier, a high N allows for too many options to have a chance of being selected.



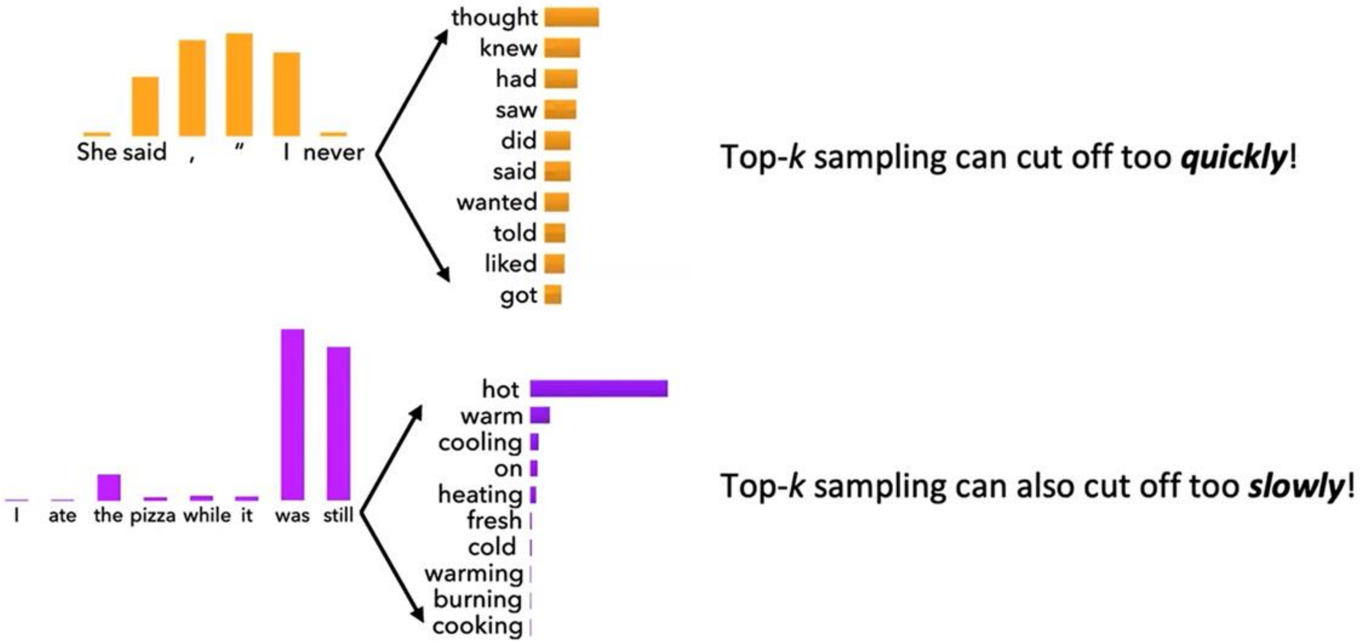
Top-*k* sampling can cut off too ***quickly***!

Top-*k* sampling can also cut off too ***slowly***!

Holtzman, Ari, et al. "The curious case of neural text degeneration." arXiv preprint arXiv:1904.09751 (2019).

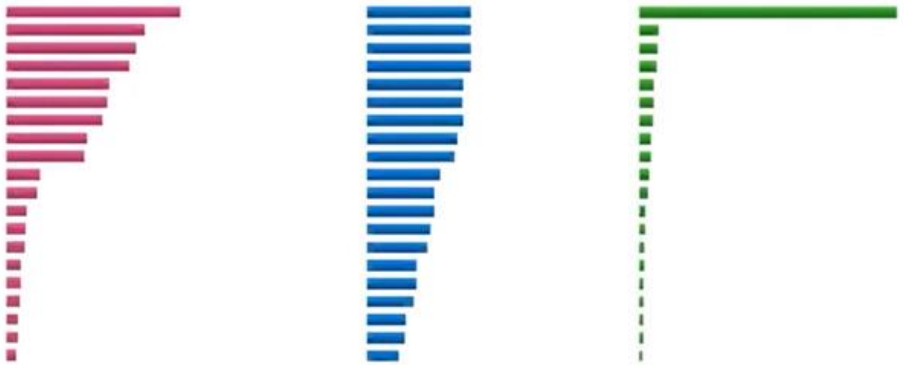
Issues with top-N sampling

- The probability distributions to sample from may be dynamic.
  - When $P^{<t>}$ is flatter, a limited N removes many viable options.
  - When $P^{<t>}$ is peakier, a high N allows for too many options to have a chance of being selected.
- Solution: Top-p sampling
  - Sample from all tokens in the top p cumulative mass (i.e. where mass is concentrated)
  - Varies N depending on the uniformity of $P^{<t>}$.



Top-*k* sampling can cut off too ***quickly***!

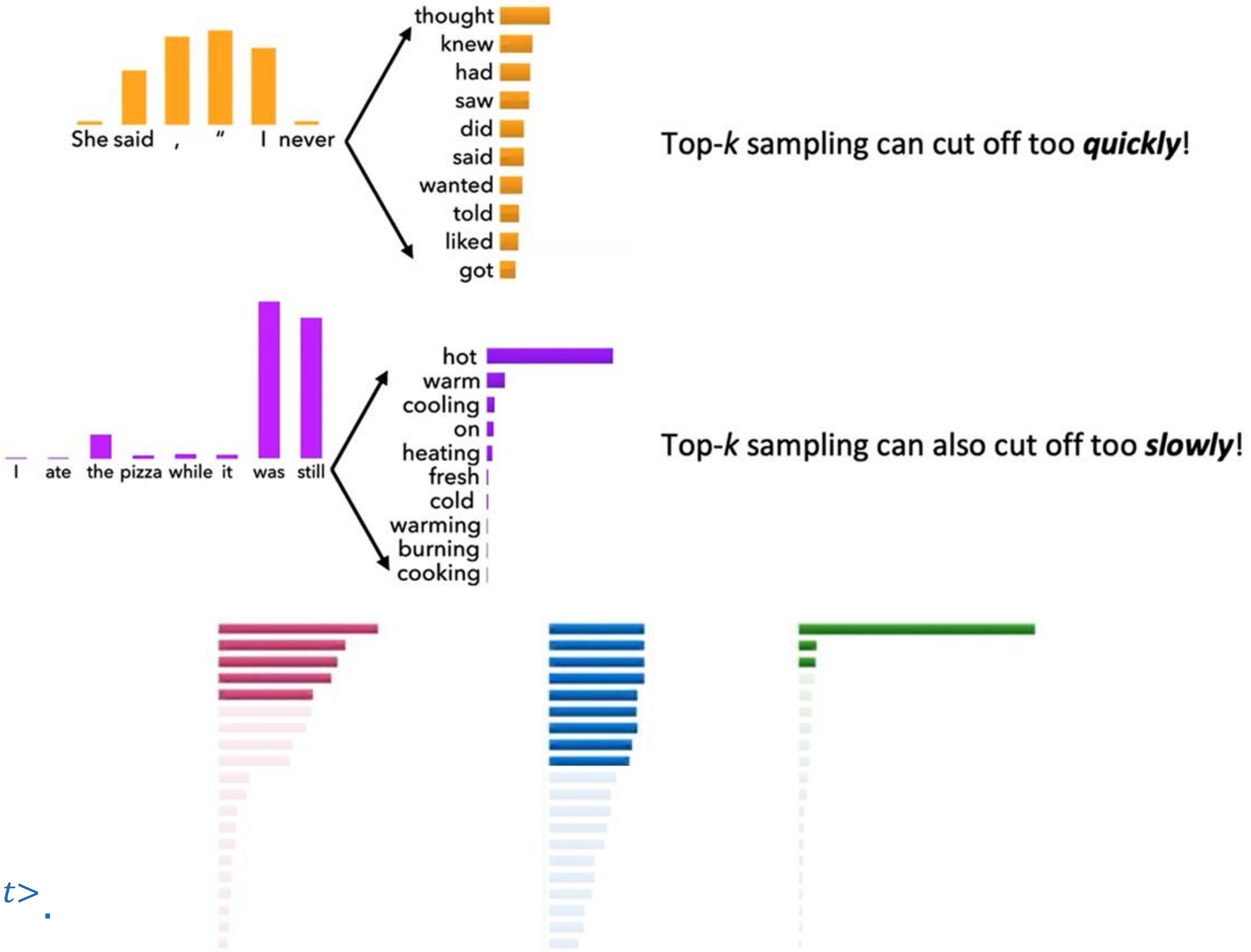
Top-*k* sampling can also cut off too ***slowly***!



Holtzman, Ari, et al. "The curious case of neural text degeneration." arXiv preprint arXiv:1904.09751 (2019).

Issues with top-N sampling

- The probability distributions to sample from may be dynamic.
  - When $P^{<t>}$ is flatter, a limited N removes many viable options.
  - When $P^{<t>}$ is peakier, a high N allows for too many options to have a chance of being selected.
- Solution: Top-p sampling
  - Sample from all tokens in the top p cumulative mass (i.e. where mass is concentrated)
  - Varies N depending on the uniformity of $P^{<t>}$.



Holtzman, Ari, et al. "The curious case of neural text degeneration." arXiv preprint arXiv:1904.09751 (2019).

$Softmax$ Temperature

- At time step t, LM computes $P^{<t>}$ by applying $softmax$ on a vector of scores $s \in \mathbb{R}^V$.

$$P^{<t>}(w) = \frac{\exp(s_w)}{\sum_{w_i \in V} \exp(s_{w_i})}$$

$Softmax$ Temperature

- At time step t, LM computes $P^{<t>}$ by applying $softmax$ on a vector of scores $s \in \mathbb{R}^V$.

$$P^{<t>}(w) = \frac{\exp(s_w)}{\sum_{w_i \in V} \exp(s_{w_i})}$$

- We can apply a temperature hyperparameter τ to $softmax$ to rebalance $P^{<t>}$.

$$P^{<t>}(w) = \frac{\exp(s_w/\tau)}{\sum_{w_i \in V} \exp(s_{w_i}/\tau)}$$

$Softmax$ Temperature

- At time step t, LM computes $P^{<t>}$ by applying $softmax$ on a vector of scores $s \in \mathbb{R}^V$.

$$P^{<t>}(w) = \frac{\exp(s_w)}{\sum_{w_i \in V} \exp(s_{w_i})}$$

- We can apply a temperature hyperparameter τ to $softmax$ to rebalance $P^{<t>}$.

$$P^{<t>}(w) = \frac{\exp(s_w/\tau)}{\sum_{w_i \in V} \exp(s_{w_i}/\tau)}$$

- Raising the temperature will melt the probability distribution.
  - $P^{<t>}$ becomes more like uniform distribution causing diverse output.

$softmax$ Temperature

- At time step t, LM computes $P^{<t>}$ by applying $softmax$ on a vector of scores $s \in \mathbb{R}^{V}$.

$$P^{<t>}(w) = \frac{\exp(s_w)}{\sum_{w_i \in V} \exp(s_{w_i})}$$

- We can apply a temperature hyperparameter τ to $softmax$ to rebalance $P^{<t>}$.

$$P^{<t>}(w) = \frac{\exp(s_w/\tau)}{\sum_{w_i \in V} \exp(s_{w_i}/\tau)}$$

- Raising the temperature will melt the probability distribution.
  - $P^{<t>}$ becomes more like uniform distribution causing diverse output.
- Lowering the temperature will shrink the probability distribution.
  - $P^{<t>}$ becomes more like a spike causing highly topic-specific output.

$softmax$ Temperature

- At time step t, LM computes $P^{<t>}$ by applying $softmax$ on a vector of scores $s \in \mathbb{R}^V$.

$$P^{<t>}(w) = \frac{\exp(s_w)}{\sum_{w_i \in V} \exp(s_{w_i})}$$

- We can apply a temperature hyperparameter τ to $softmax$ to rebalance $P^{<t>}$.

$$P^{<t>}(w) = \frac{\exp(s_w/\tau)}{\sum_{w_i \in V} \exp(s_{w_i}/\tau)}$$

- Raising the temperature will melt the probability distribution.
  - $P^{<t>}$ becomes more like uniform distribution causing diverse output.
- Lowering the temperature will shrink the probability distribution.
  - $P^{<t>}$ becomes more like a spike causing highly topic-specific output.

Temperature-scaled $softmax$ is not a decoding algorithm

It's a technique you can apply at test time, in conjunction with a decoding algorithm.

Summarisation compresses input length while preserving information

- Given input text x, generate a text y that is shorter in length than x and contains the same information as x.

https://github.com/mathsyouth/awesome-text-summarization

Summarisation compresses input length while preserving information

- Given input text x, generate a text y that is shorter in length than x and contains the same information as x.

- Single-document: Generate y based on x

  - Various datasets of different lengths and styles exist.

    - Gigaword: Write a news headline based on first one or two sentences.
    - NYT/CNN/DailyMail: Write multi-sentence summary of a news article.
    - Wikihow: Wite multi-sentence summary of full how-to articles.

https://github.com/mathsyouth/awesome-text-summarization

Summarisation compresses input length while preserving information

- Given input text x, generate a text y that is shorter in length than x and contains the same information as x.

- Single-document: Generate y based on x

  - Various datasets of different lengths and styles exist.

    - Gigaword: Write a news headline based on first one or two sentences.
    - NYT/CNN/DailyMail: Write multi-sentence summary of a news article.
    - Wikihow: Wite multi-sentence summary of full how-to articles.

- Multi-document: Generate a single y based on $x_1, x_2, \dots, x_n$

  - $x_1, x_2, \dots, x_n$ may have overlapping contents. For example, different news articles of the same event.

https://github.com/mathsyouth/awesome-text-summarization

Summarisation compresses input length while preserving information

- Given input text x, generate a text y that is shorter in length than x and contains the same information as x.

- Single-document: Generate y based on x

  - Various datasets of different lengths and styles exist.

    - Gigaword: Write a news headline based on first one or two sentences.
    - NYT/CNN/DailyMail: Write multi-sentence summary of a news article.
    - Wikihow: Wite multi-sentence summary of full how-to articles.

- Multi-document: Generate a single y based on $x_1, x_2, \dots, x_n$

  - $x_1, x_2, \dots, x_n$ may have overlapping contents. For example, different news articles of the same event.

- Text simplification: Rewrite the text in simpler (word choices / sentence structure) and possibly fewer words.

  - Simple Wikipedia: Standard Wikipedia to Simple Version

https://github.com/mathsyouth/awesome-text-summarization

Summarisation could be abstractive or extractive

- Extractive Summarisation
  - Identify and select important parts of original text (sentences, phrases, or words) to compile new text.
  - Usually unsupervised.
  - Is easier but restrictive

Graphic art

A type of fine art, graphic art co broad range of art forms.

Graphic art is typically and includes calligra drawing, painting, p phy, typography, s een printing), and bi

Graphic art also consists of drawn plans and layouts for interior and architectural designs.

Summarisation could be abstractive or extractive

- Extractive Summarisation
  - Identify and select important parts of original text (sentences, phrases, or words) to compile new text.
  - Usually unsupervised.
  - Is easier but restrictive
- Abstractive Summarisation
  - Learn the crux of the matter and rephrase using NLG techniques.
  - Requires Deep Learning (in 2022).
  - Difficult but flexible

Graphic art

A type of fine art, graphic art co broad range of art forms.

Graphic art is typically and includes calligra drawing, painting, p phy, typography, s een printing), and bi

Graphic art also consists of drawn plans and layouts for interior and architectural designs.

Pre-Neural Summarisation was mostly extractive

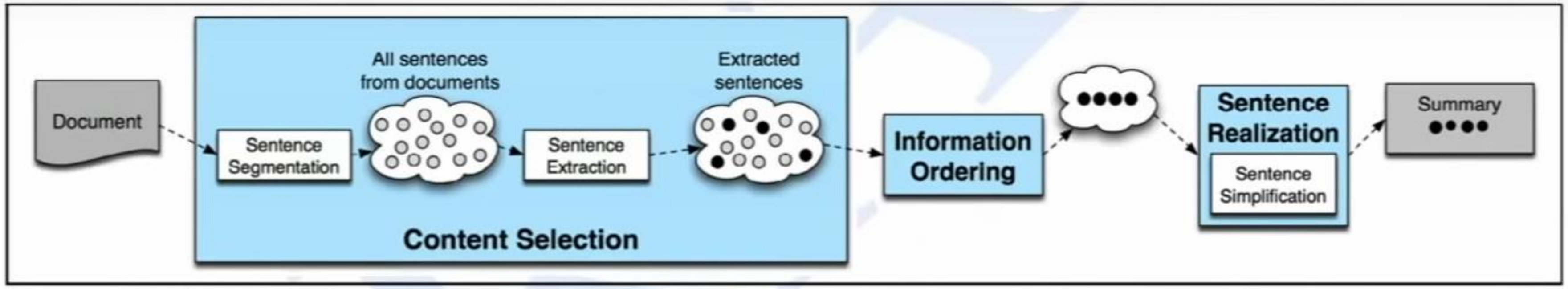


Figure 23.14 The basic architecture of a generic single document summarizer.

Pre-Neural Summarisation was mostly extractive

- The pipeline consisted of
  - Content Selection: Which sentences/word/phrases to include?
  - Information Ordering: Decide the order of the selected content.
  - Sentence Realisation: Edit the sequence of sentence and make necessary tweaks.
    - Simplify, Prune, Maintain Continuity etc.

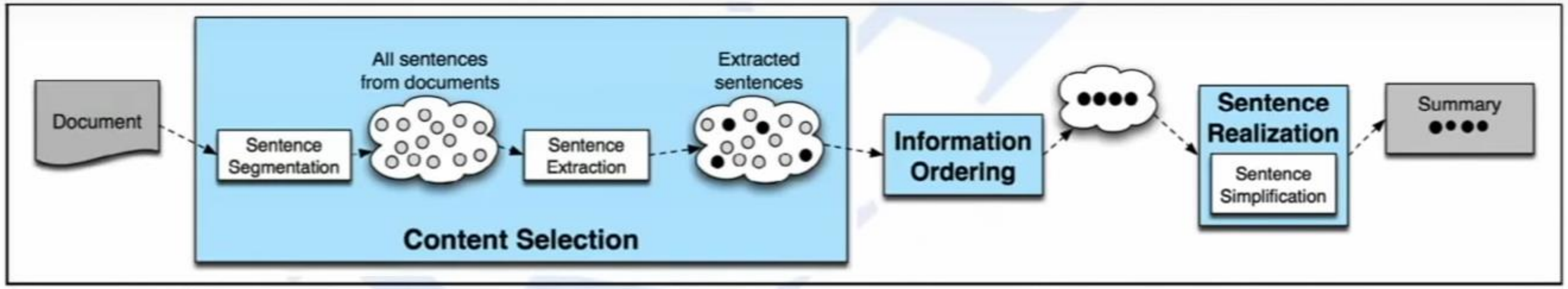
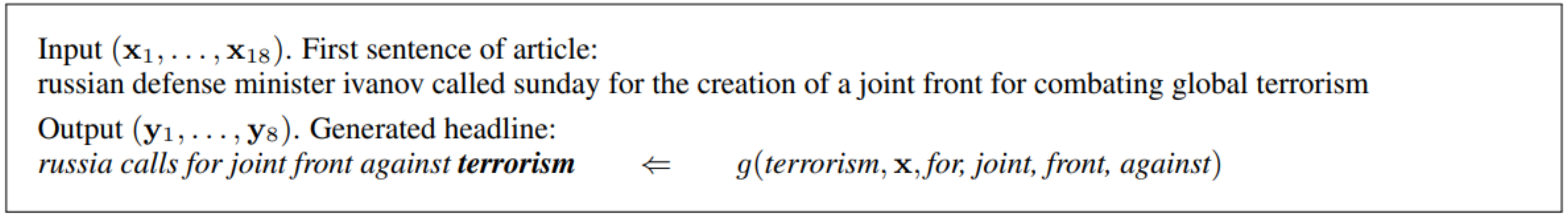


Figure 23.14 The basic architecture of a generic single document summarizer.

Neural Summarisation

- Rush et al. presented the first $seq2seq$ model for abstractive summarisation in 2015.
  - Considered single document summarisation as a translation task.
  - Used Gigaword dataset.



Input $(\mathbf{x}_1, \ldots, \mathbf{x}_{18})$. First sentence of article:
russian defense minister ivanov called sunday for the creation of a joint front for combating global terrorism

Output $(\mathbf{y}_1, \ldots, \mathbf{y}_8)$. Generated headline:
russia calls for joint front against ***terrorism*** $\Leftarrow$ $g(\textit{terrorism}, \mathbf{x}, \textit{for, joint, front, against})$

Figure 2: Example input sentence and the generated summary. The score of generating $\mathbf{y}_{i+1}$ (`terrorism`) is based on the context $\mathbf{y}_c$ (`for ... against`) as well as the input $\mathbf{x}_1 \ldots \mathbf{x}_{18}$. Note that the summary generated is abstractive which makes it possible to *generalize* (`russian defense minister` to `russia`) and *paraphrase* (`for combating` to `against`), in addition to *compressing* (dropping `the creation of`), see Jing (2002) for a survey of these editing operations.

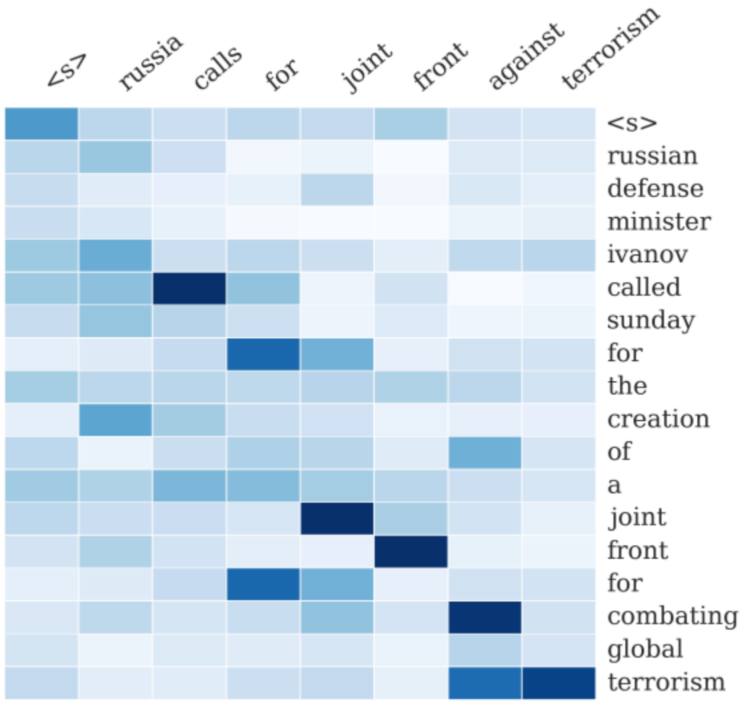


Figure 1: Example output of the attention-based summarization (ABS) system. The heatmap represents a soft alignment between the input (right) and the generated summary (top). The columns represent the distribution over the input after generating each word.

Rush, Alexander M., Sumit Chopra, and Jason Weston. "A neural attention model for abstractive sentence summarization." arXiv preprint arXiv:1509.00685 (2015).

ROUGE is used for evaluating summarisation

- ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation.

ROUGE, Lin CY. "A package for automatic evaluation of summaries." Proceedings of Workshop on Text Summarization of ACL, Spain. 2004.
https://www.youtube.com/watch?v=TMshhnrEXlg

ROUGE is used for evaluating summarisation

- ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation.

- BLEU is based on n-gram precision (why?), ROUGE-N is based on n-gram recall (why?).

ROUGE, Lin CY. "A package for automatic evaluation of summaries." Proceedings of Workshop on Text Summarization of ACL, Spain. 2004.
https://www.youtube.com/watch?v=TMshhnrEXlg

ROUGE is used for evaluating summarisation

- ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation.

- BLEU is based on n-gram precision (why?), ROUGE-N is based on n-gram recall (why?).

- BLEU enforces a brevity penalty (why?), ROUGE does not enforce any such penalty (why?).

ROUGE, Lin CY. "A package for automatic evaluation of summaries." Proceedings of Workshop on Text Summarization of ACL, Spain. 2004.
https://www.youtube.com/watch?v=TMshhnrEXlg

ROUGE is used for evaluating summarisation

- ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation.

- BLEU is based on n-gram precision (why?), ROUGE-N is based on n-gram recall (why?).

- BLEU enforces a brevity penalty (why?), ROUGE does not enforce any such penalty (why?).

- BLEU is a single number which incorporates multiple n-grams overlaps, ROUGE is reported separately for each n-gram (ROUGE-1, ROUGE-2 etc.).

ROUGE, Lin CY. "A package for automatic evaluation of summaries." Proceedings of Workshop on Text Summarization of ACL, Spain. 2004.
https://www.youtube.com/watch?v=TMshhnrEXlg

ROUGE is used for evaluating summarisation

- ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation.

- BLEU is based on n-gram precision (why?), ROUGE-N is based on n-gram recall (why?).

- BLEU enforces a brevity penalty (why?), ROUGE does not enforce any such penalty (why?).

- BLEU is a single number which incorporates multiple n-grams overlaps, ROUGE is reported separately for each n-gram (ROUGE-1, ROUGE-2 etc.).

- ROUGE is normally supplemented by $F-1$ scores because of not explicit MAX_LEN constraint.

ROUGE, Lin CY. "A package for automatic evaluation of summaries." Proceedings of Workshop on Text Summarization of ACL, Spain. 2004.
https://www.youtube.com/watch?v=TMshhnrEXlg

ROUGE is used for evaluating summarisation

- ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation.

- BLEU is based on n-gram precision (why?), ROUGE-N is based on n-gram recall (why?).

- BLEU enforces a brevity penalty (why?), ROUGE does not enforce any such penalty (why?).

- BLEU is a single number which incorporates multiple n-grams overlaps, ROUGE is reported separately for each n-gram (ROUGE-1, ROUGE-2 etc.).

- ROUGE is normally supplemented by $F-1$ scores because of not explicit MAX_LEN constraint.

- ROUGE-L looks for Longest Common Subsequence in the summary instead of n-gram recall.

$$ROUGE-L = \frac{LCS}{Number\ of\ words\ in\ reference}$$

ROUGE, Lin CY. "A package for automatic evaluation of summaries." Proceedings of Workshop on Text Summarization of ACL, Spain. 2004.
https://www.youtube.com/watch?v=TMshhnrEXlg

Dialogue generation has many varieties

- Task-Oriented Dialogues
  - Assistive dialogue generation:
    - Customer services, recommendations, question answering etc.
    - Siri and Alexa are examples of such dialogue bots.
  - Cooperative dialogue generation:
    - Two agents cooperate to solve a task.
  - Adversarial dialogue generation:
    - Two agents compete in a task.

Dialogue generation has many varieties

- Task-Oriented Dialogues
    - Assistive dialogue generation:
        - Customer services, recommendations, question answering etc.
        - Siri and Alexa are examples of such dialogue bots.
    - Cooperative dialogue generation:
        - Two agents cooperate to solve a task.
    - Adversarial dialogue generation:
        - Two agents compete in a task.
- Social Dialogues:
    - Chit-chat for fun or company or therapy.

Pre and post DL dialogue generation

- Pre DL dialogue generation was restricted to template-based or retrieval-based sentence generation.
    - Because of difficulty in freeform NLG.
    - Natural Language Understanding was still a part of such pre-neural dialogue generation systems.

Useful resource: *https://medium.com/x8-the-ai-community/a-reading-list-and-mini-survey-of-conversational-ai-32fceea97180*

Pre and post DL dialogue generation

- Pre DL dialogue generation was restricted to template-based or retrieval-based sentence generation.
  - Because of difficulty in freeform NLG.
  - Natural Language Understanding was still a part of such pre-neural dialogue generation systems.
- Open-ended freeform dialogue generation using $seq2seq$ emerged in 2015.

Useful resource: *https://medium.com/x8-the-ai-community/a-reading-list-and-mini-survey-of-conversational-ai-32fceea97180*

Pre and post DL dialogue generation

- Pre DL dialogue generation was restricted to template-based or retrieval-based sentence generation.
  - Because of difficulty in freeform NLG.
  - Natural Language Understanding was still a part of such pre-neural dialogue generation systems.
- Open-ended freeform dialogue generation using $seq2seq$ emerged in 2015.
- However, it quickly became apparent that naïve application of $seq2seq$ models for dialogue generation is not very helpful.
  - Generic/Boring, Irrelevant, Repetitive (within or across dialogues), Lack of context, inconsistent persona.

Useful resource: *https://medium.com/x8-the-ai-community/a-reading-list-and-mini-survey-of-conversational-ai-32fceea97180*

How to fix generic response?

- Late interventions (at test time).
  - Use sampling decoding algorithms.
  - Use $softmax$ temperature.
  - Upweight rare words during beam search.

Jiang, Shaojie, and Maarten de Rijke. "Why are sequence-to-sequence models so dull? understanding the low-diversity problem of chatbots." arXiv preprint arXiv:1809.01941 (2018).

How to fix generic response?

- Late interventions (at test time).
  - Use sampling decoding algorithms.
  - Use $softmax$ temperature.
  - Upweight rare words during beam search.
- Early interventions (during training)
  - Condition the decoder on additional content from the source.
    - Sample some words relevant to S and attend to them during decoding.
  - Train a retrieve-and-refine model instead of train-from-scratch model.
    - Produces more diverse, human-like, interesting dialogues.

Jiang, Shaojie, and Maarten de Rijke. "Why are sequence-to-sequence models so dull? understanding the low-diversity problem of chatbots." arXiv preprint arXiv:1809.01941 (2018).

How to fix repetition in response?

- Simple solution is to block repeating n-grams during beam search.
  - Usually quite effective for avoiding repetition within a dialogue.

See, A., Liu, P. J., & Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. arXiv preprint arXiv:1704.04368.
Welleck, Sean, et al. "Neural text generation with unlikelihood training." arXiv preprint arXiv:1908.04319 (2019).

How to fix repetition in response?

- Simple solution is to block repeating n-grams during beam search.
    - Usually quite effective for avoiding repetition within a dialogue.
- Train using coverage loss.
    - An objective that prevents attention from attending to words that have already been covered (picked).
        - Assumes that repetition is because of repeated attention.

See, A., Liu, P. J., & Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. arXiv preprint arXiv:1704.04368.
Welleck, Sean, et al. "Neural text generation with unlikelihood training." arXiv preprint arXiv:1908.04319 (2019).

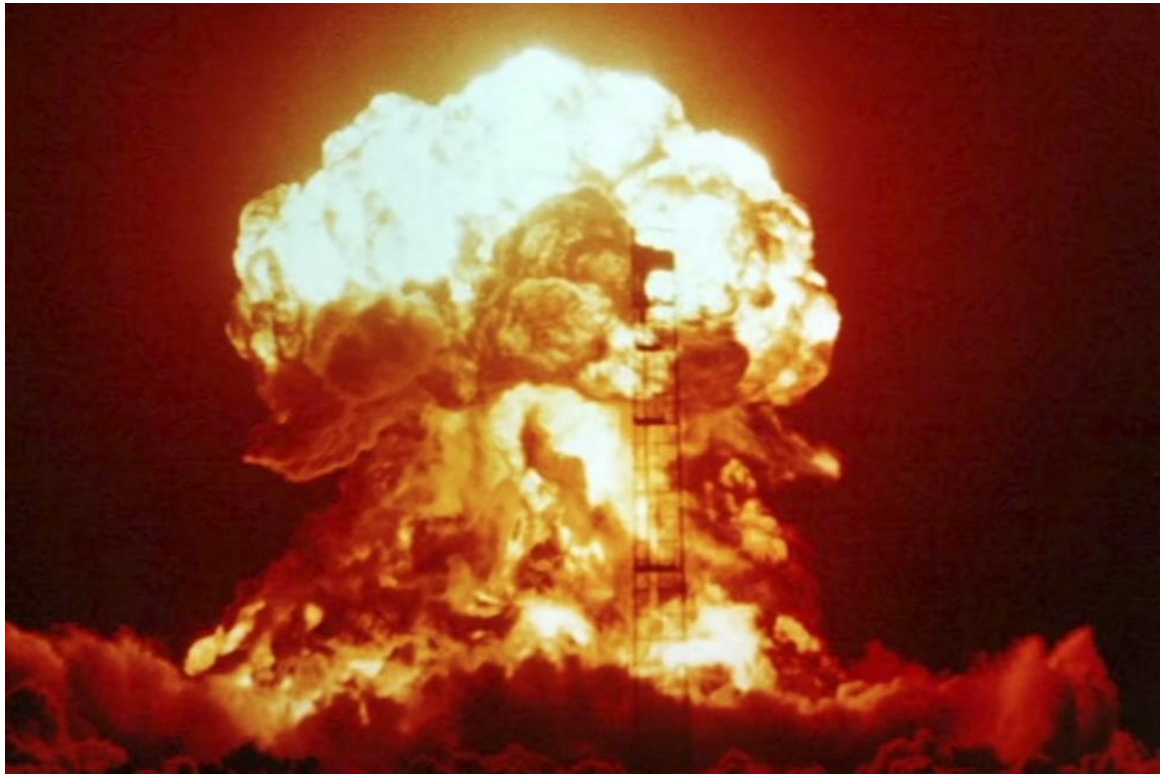
How to fix repetition in response?

- Simple solution is to block repeating n-grams during beam search.
  - Usually quite effective for avoiding repetition within a dialogue.
- Train using coverage loss.
  - An objective that prevents attention from attending to words that have already been covered (picked).
    - Assumes that repetition is because of repeated attention.
- Unlikelihood Objective
  - Penalise generation of already-seen tokens.

See, A., Liu, P. J., & Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. arXiv preprint arXiv:1704.04368.
Welleck, Sean, et al. "Neural text generation with unlikelihood training." arXiv preprint arXiv:1908.04319 (2019).

Storytelling is more open ended NLG task

You have to be the only light bulb in the night sky. I thought, Oh, God, it's so dark out of me that I missed you, I promise.



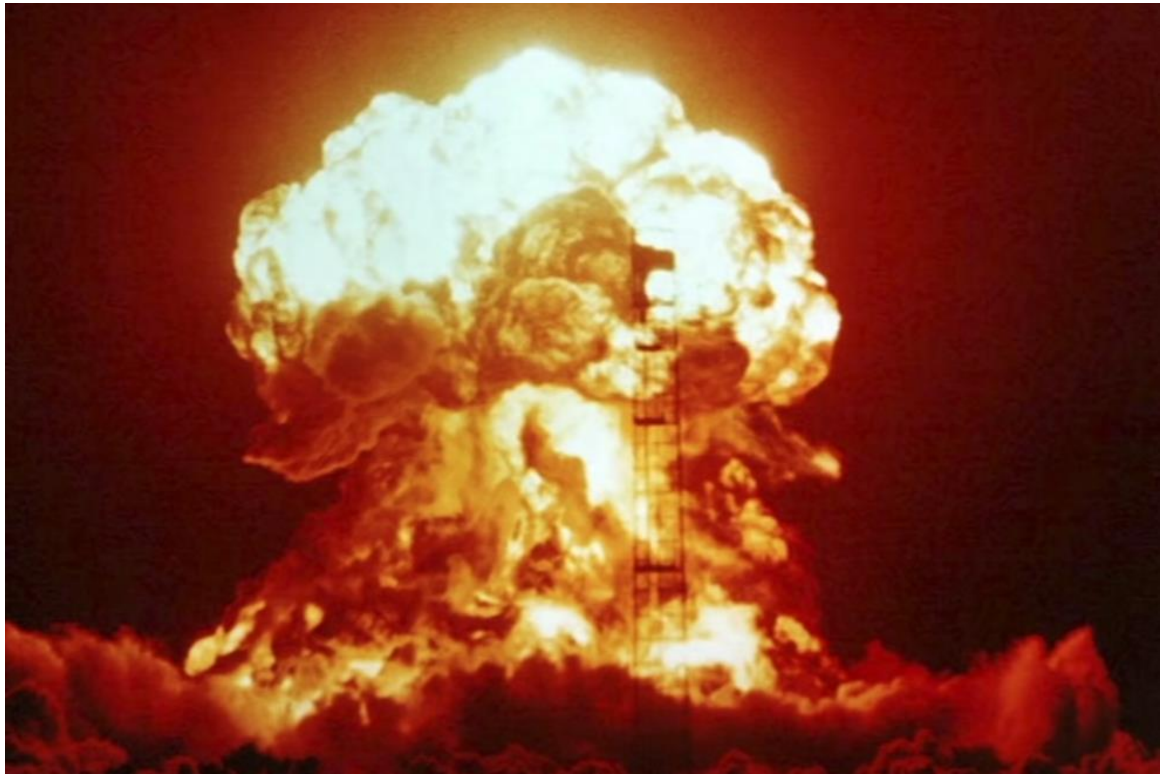
Workshop Link: *http://www.visionandlanguage.net/workshop2019/*

Interesting Resource: *https://medium.com/@samim/generating-stories-about-images-d163ba41e4ed*

Storytelling is more open ended NLG task

- Most storytelling models require some sort of prompt. The prompt could be,
  - An images.
  - First few lines of the story or keywords.
  - Story written so far.
    - Write the next sentence.

You have to be the only light bulb in the night sky. I thought, Oh, God, it's so dark out of me that I missed you, I promise.



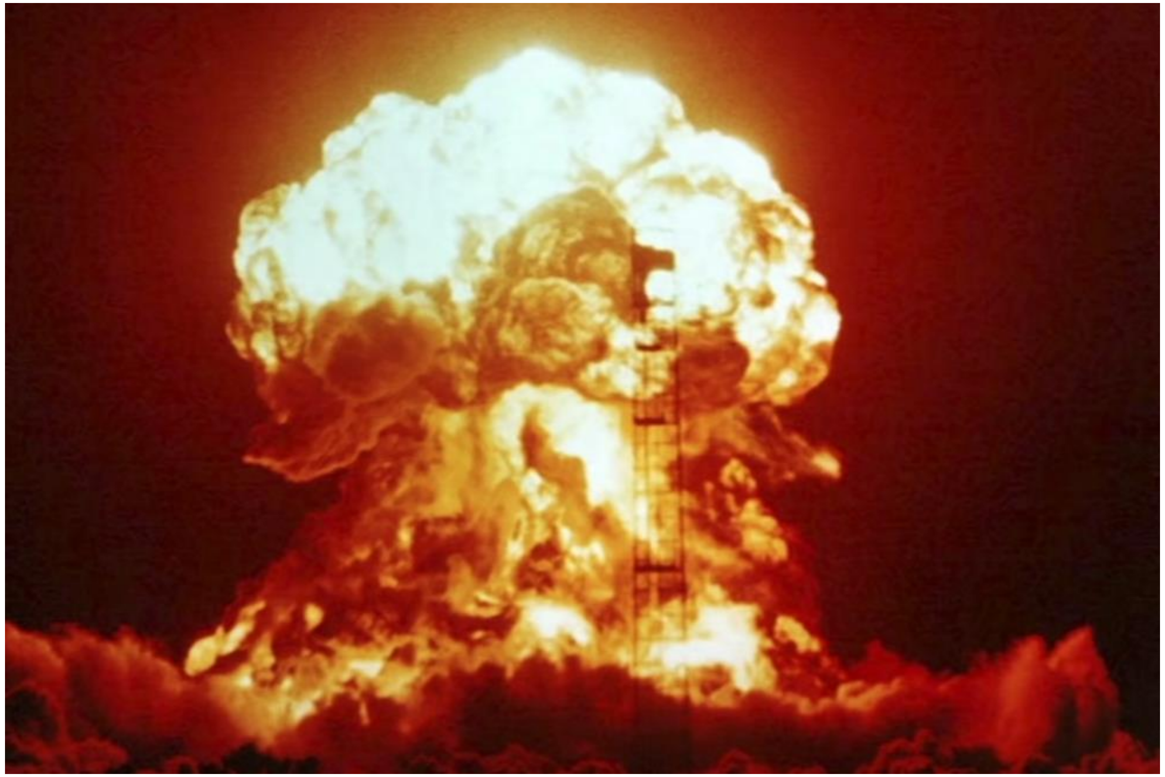
Workshop Link: *http://www.visionandlanguage.net/workshop2019/*

Interesting Resource: *https://medium.com/@samim/generating-stories-about-images-d163ba41e4ed*

Storytelling is more open ended NLG task

- Most storytelling models require some sort of prompt. The prompt could be,
  - An images.
  - First few lines of the story or keywords.
  - Story written so far.
    - Write the next sentence.
- First story telling workshop was held in 2018.
  - Held a competition to write story accompanying a sequence of five images.

You have to be the only light bulb in the night sky. I thought, Oh, God, it's so dark out of me that I missed you, I promise.

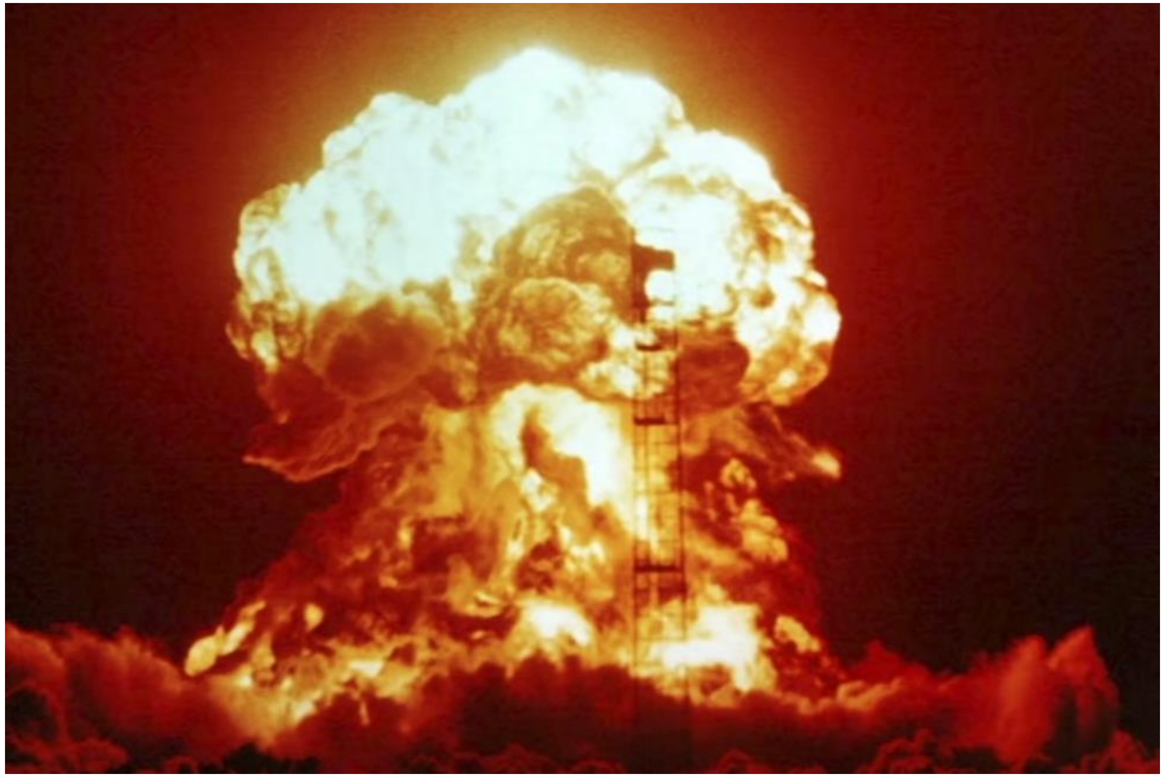


Workshop Link: *http://www.visionandlanguage.net/workshop2019/*
Interesting Resource: *https://medium.com/@samim/generating-stories-about-images-d163ba41e4ed*

Storytelling is more open ended NLG task

- Most storytelling models require some sort of prompt. The prompt could be,
  - An images.
  - First few lines of the story or keywords.
  - Story written so far.
    - Write the next sentence.
- First story telling workshop was held in 2018.
  - Held a competition to write story accompanying a sequence of five images.
- Storytelling using images is not like image captioning.
  - No image-story paired data to learn from.

You have to be the only light bulb in the night sky. I thought, Oh, God, it's so dark out of me that I missed you, I promise.



Workshop Link: *http://www.visionandlanguage.net/workshop2019/*

Interesting Resource: *https://medium.com/@samim/generating-stories-about-images-d163ba41e4ed*

Generating a story from an image is tricky

- How to get around lack of parallel data?
    - Use a common-sentence encoding space. What the hell is that?
    - An encoding space that is common between Image Processing Model and Language Model.

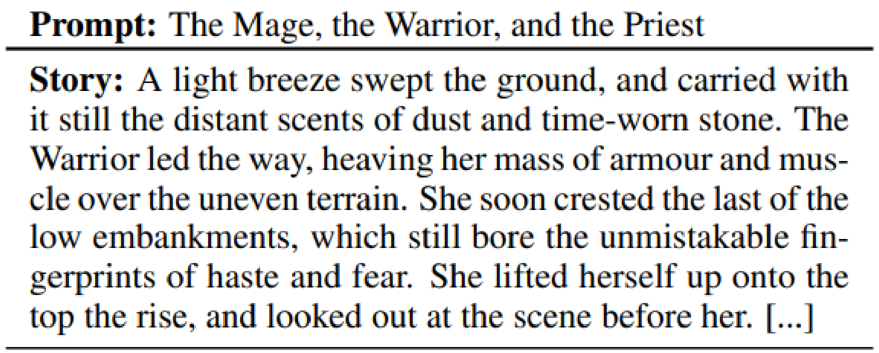
Kiros, Ryan, et al. "Skip-thought vectors." Advances in neural information processing systems 28 (2015).

Generating a story from an image is tricky

- How to get around lack of parallel data?
  - Use a common-sentence encoding space. What the hell is that?
  - An encoding space that is common between Image Processing Model and Language Model.
- Skip-Thought Vectors are a type of unsupervised general-purpose sentence encoding methods.
  - Learn the embedding of a sentence using its neighbouring sentences.
  - Used Microsoft COCO dataset to learn a mapping from images to skip-thought encoding of their captions.
  - Separately, used a target styled corpus to train an RNN-based Language Model.
  - Put the two model together sharing the encoding space and enjoy!

Kiros, Ryan, et al. "Skip-thought vectors." Advances in neural information processing systems 28 (2015).

Generating a story from a writing prompt



Prompt: The Mage, the Warrior, and the Priest

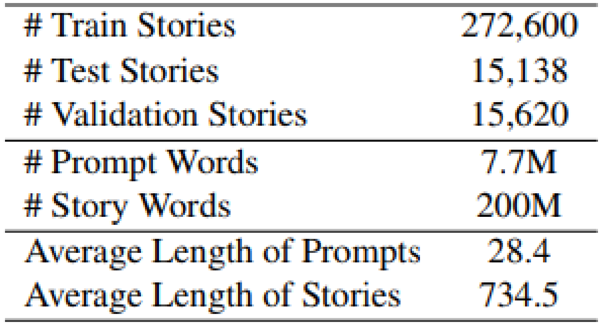
Story: A light breeze swept the ground, and carried with it still the distant scents of dust and time-worn stone. The Warrior led the way, heaving her mass of armour and muscle over the uneven terrain. She soon crested the last of the low embankments, which still bore the unmistakable fingerprints of haste and fear. She lifted herself up onto the top the rise, and looked out at the scene before her. [...]

Fan, Angela, Mike Lewis, and Yann Dauphin. "Hierarchical neural story generation." arXiv preprint arXiv:1805.04833 (2018).

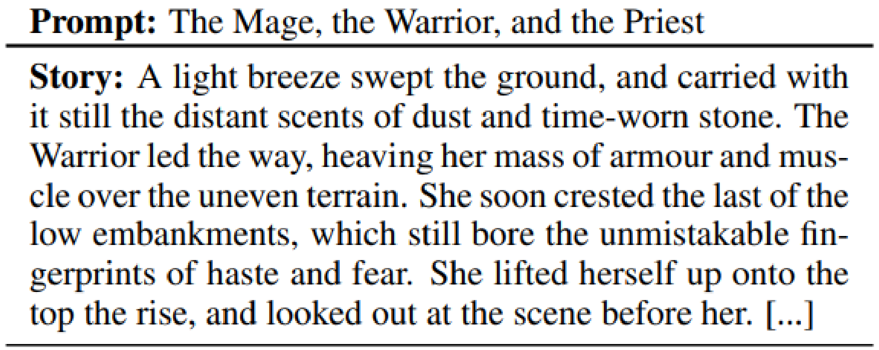
Generating a story from a writing prompt

- Fan et al. released a new dataset in 2018 for story generation collected from Reddit’s WritingPrompts subreddit.

- They also developed a CNN based $seq2seq$ model for prompt-to-story generation.



# Train Stories	272,600
# Test Stories	15,138
# Validation Stories	15,620
# Prompt Words	7.7M
# Story Words	200M
Average Length of Prompts	28.4
Average Length of Stories	734.5



Prompt: The Mage, the Warrior, and the Priest

Story: A light breeze swept the ground, and carried with it still the distant scents of dust and time-worn stone. The Warrior led the way, heaving her mass of armour and muscle over the uneven terrain. She soon crested the last of the low embankments, which still bore the unmistakable fingerprints of haste and fear. She lifted herself up onto the top the rise, and looked out at the scene before her. [...]

Fan, Angela, Mike Lewis, and Yann Dauphin. "Hierarchical neural story generation." arXiv preprint arXiv:1805.04833 (2018).

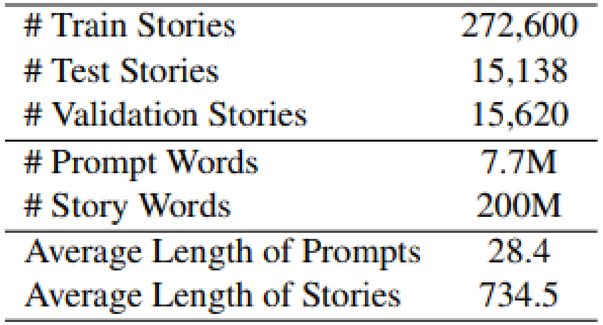
Generating a story from a writing prompt

- Fan et al. released a new dataset in 2018 for story generation collected from Reddit’s WritingPrompts subreddit.

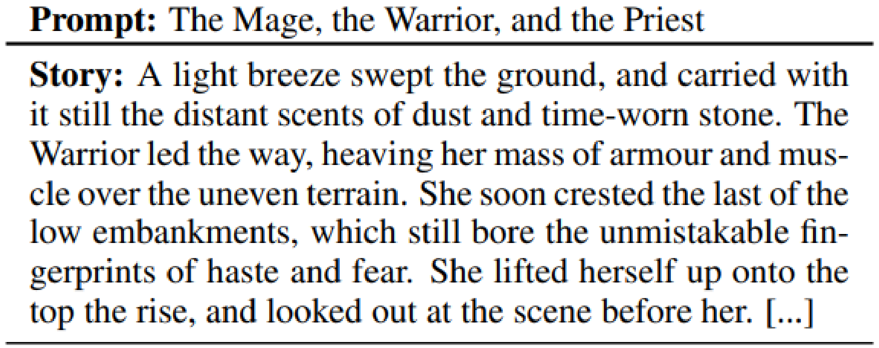
- They also developed a CNN based $seq2seq$ model for prompt-to-story generation.

- Used Gated Multi-head Multi-scale self-attention.

  - Self Attention helps maintain long-range context.
  - Gates regulate how selective attention mechanism should be.
  - Different attention heads attend at different scales
    - Fine grained details versus coarse-grained details.



# Train Stories	272,600
# Test Stories	15,138
# Validation Stories	15,620
# Prompt Words	7.7M
# Story Words	200M
Average Length of Prompts	28.4
Average Length of Stories	734.5



Prompt: The Mage, the Warrior, and the Priest

Story: A light breeze swept the ground, and carried with it still the distant scents of dust and time-worn stone. The Warrior led the way, heaving her mass of armour and muscle over the uneven terrain. She soon crested the last of the low embankments, which still bore the unmistakable fingerprints of haste and fear. She lifted herself up onto the top the rise, and looked out at the scene before her. [...]

Fan, Angela, Mike Lewis, and Yann Dauphin. "Hierarchical neural story generation." arXiv preprint arXiv:1805.04833 (2018).

Generating a story from a writing prompt

- Fan et al. released a new dataset in 2018 for story generation collected from Reddit’s WritingPrompts subreddit.

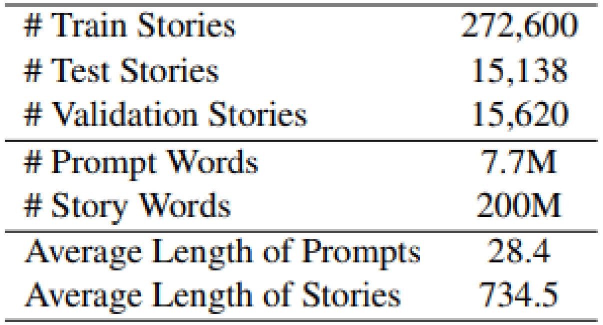
- They also developed a CNN based $seq2seq$ model for prompt-to-story generation.

- Used Gated Multi-head Multi-scale self-attention.

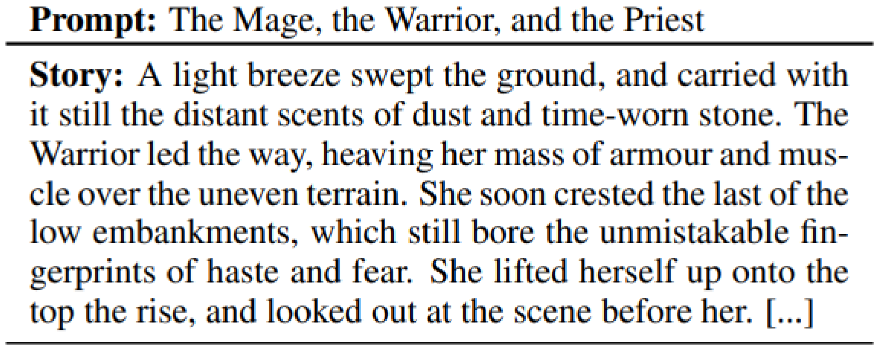
  - Self Attention helps maintain long-range context.
  - Gates regulate how selective attention mechanism should be.
  - Different attention heads attend at different scales
    - Fine grained details versus coarse-grained details.

- There are two $seq2seq$ models.

  - A convolutional language model learns the sketch of the story.
  - Another seq2seq language models learns to condition on the sketch. Consistency beyond single phrase.



# Train Stories	272,600
# Test Stories	15,138
# Validation Stories	15,620
# Prompt Words	7.7M
# Story Words	200M
Average Length of Prompts	28.4
Average Length of Stories	734.5



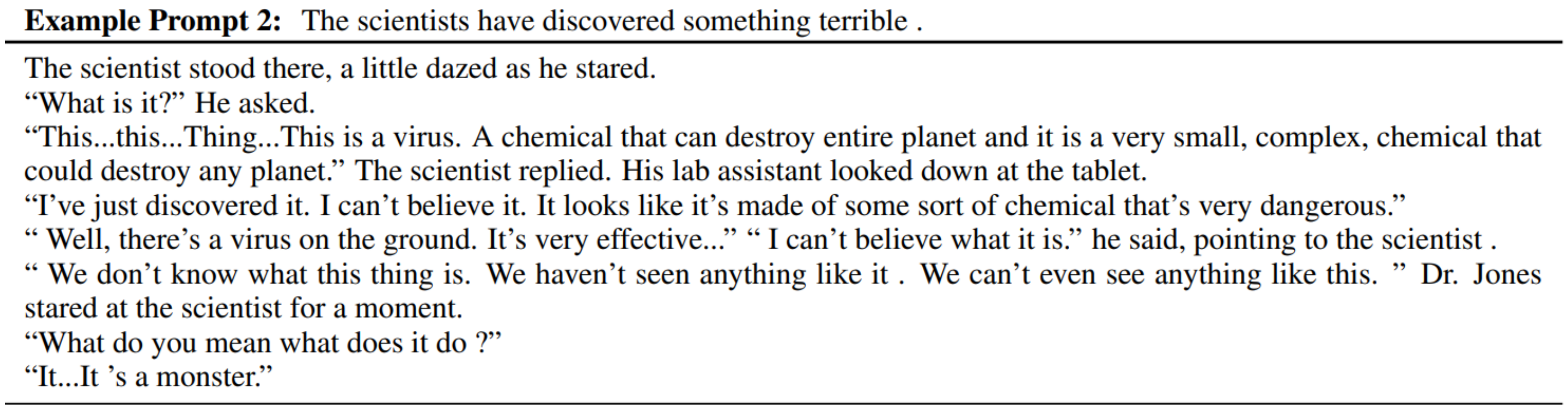
Prompt: The Mage, the Warrior, and the Priest

Story: A light breeze swept the ground, and carried with it still the distant scents of dust and time-worn stone. The Warrior led the way, heaving her mass of armour and muscle over the uneven terrain. She soon crested the last of the low embankments, which still bore the unmistakable fingerprints of haste and fear. She lifted herself up onto the top the rise, and looked out at the scene before her. [...]

Fan, Angela, Mike Lewis, and Yann Dauphin. "Hierarchical neural story generation." arXiv preprint arXiv:1805.04833 (2018).

The results of prompt-to-story are impressive

- The results are related to prompt, diverse, and stylistically dramatic.



Example Prompt 2: The scientists have discovered something terrible .

The scientist stood there, a little dazed as he stared.
“What is it?” He asked.
“This...this...Thing...This is a virus. A chemical that can destroy entire planet and it is a very small, complex, chemical that could destroy any planet.” The scientist replied. His lab assistant looked down at the tablet.
“I’ve just discovered it. I can’t believe it. It looks like it’s made of some sort of chemical that’s very dangerous.”
“ Well, there’s a virus on the ground. It’s very effective...” “ I can’t believe what it is.” he said, pointing to the scientist .
“ We don’t know what this thing is. We haven’t seen anything like it . We can’t even see anything like this. ” Dr. Jones stared at the scientist for a moment.
“What do you mean what does it do ?”
“It...It ’s a monster.”

Fan, Angela, Mike Lewis, and Yann Dauphin. "Hierarchical neural story generation." arXiv preprint arXiv:1805.04833 (2018).

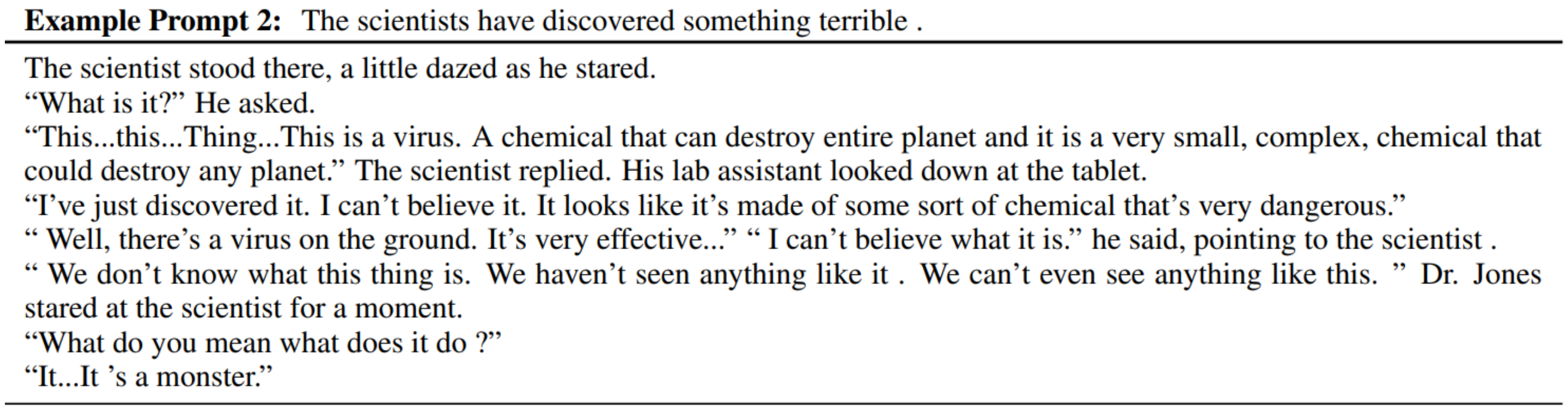
The results of prompt-to-story are impressive

- The results are related to prompt, diverse, and stylistically dramatic.

- But they are also mostly,

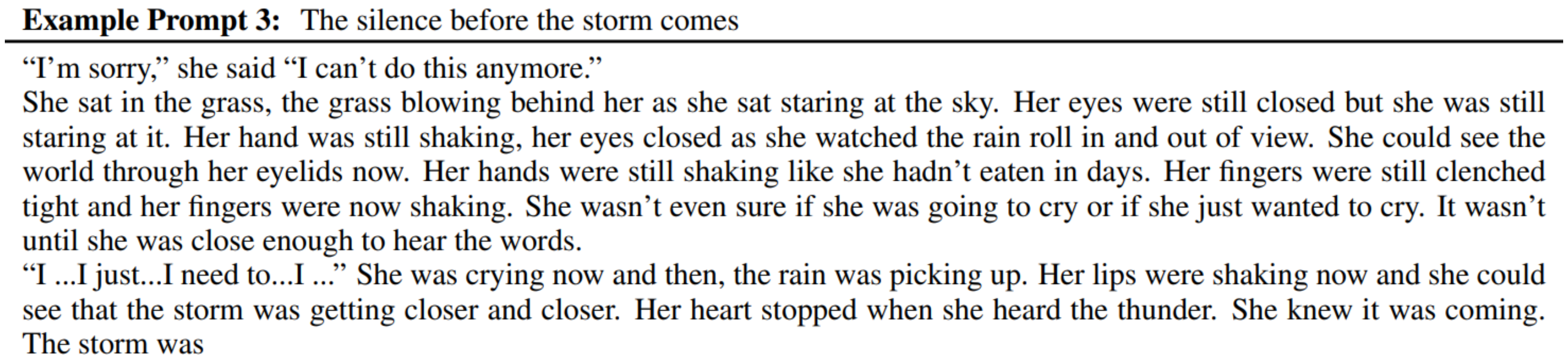
  - Descriptive: focus on just scene-setting.

  - Stagnant on the same idea when generating longer sequences.



Example Prompt 2: The scientists have discovered something terrible .

The scientist stood there, a little dazed as he stared.
“What is it?” He asked.
“This...this...Thing...This is a virus. A chemical that can destroy entire planet and it is a very small, complex, chemical that could destroy any planet.” The scientist replied. His lab assistant looked down at the tablet.
“I’ve just discovered it. I can’t believe it. It looks like it’s made of some sort of chemical that’s very dangerous.”
“ Well, there’s a virus on the ground. It’s very effective...” “ I can’t believe what it is.” he said, pointing to the scientist .
“ We don’t know what this thing is. We haven’t seen anything like it . We can’t even see anything like this. ” Dr. Jones stared at the scientist for a moment.
“What do you mean what does it do ?”
“It...It ’s a monster.”



Example Prompt 3: The silence before the storm comes

“I’m sorry,” she said “I can’t do this anymore.”
She sat in the grass, the grass blowing behind her as she sat staring at the sky. Her eyes were still closed but she was still staring at it. Her hand was still shaking, her eyes closed as she watched the rain roll in and out of view. She could see the world through her eyelids now. Her hands were still shaking like she hadn’t eaten in days. Her fingers were still clenched tight and her fingers were now shaking. She wasn’t even sure if she was going to cry or if she just wanted to cry. It wasn’t until she was close enough to hear the words.
“I ...I just...I need to...I ...” She was crying now and then, the rain was picking up. Her lips were shaking now and she could see that the storm was getting closer and closer. Her heart stopped when she heard the thunder. She knew it was coming. The storm was

Fan, Angela, Mike Lewis, and Yann Dauphin. "Hierarchical neural story generation." arXiv preprint arXiv:1805.04833 (2018).

Storytelling is a really challenging task, even for humans

- Neural story generators can sound fluent but they are also meandering and lack plot coherence.

Storytelling is a really challenging task, even for humans

- Neural story generators can sound fluent but they are also meandering and lack plot coherence.

- Why is neural storytelling so difficult?

    - Language Models work well with sequence of words, storytelling requires sequence of events.

Storytelling is a really challenging task, even for humans

- Neural story generators can sound fluent but they are also meandering and lack plot coherence.

- Why is neural storytelling so difficult?

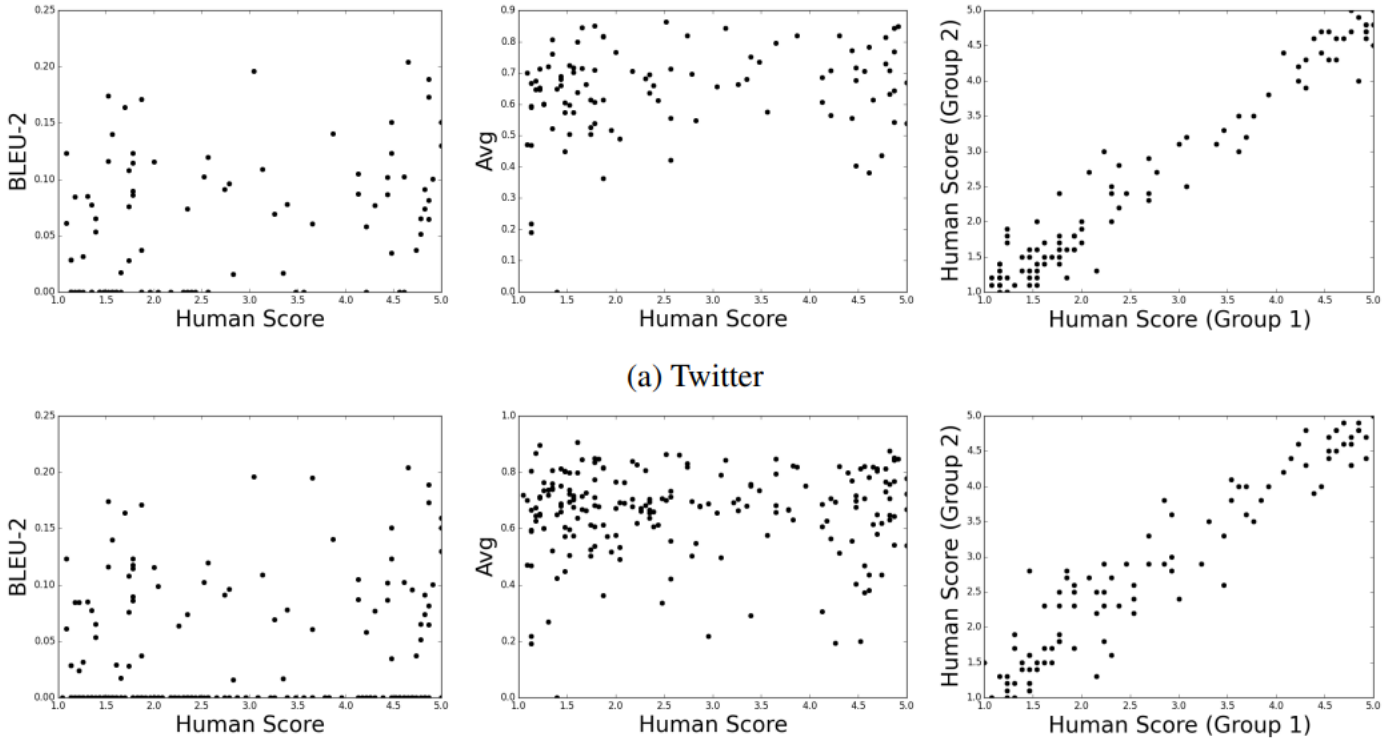
    - Language Models work well with sequence of words, storytelling requires sequence of events.

- A Good storytelling model should understand and model,

    - Events and causality structure between them
    - Characters, their personalities, backgrounds, motivations, and relationships to other characters.
    - State of the world in the story.
    - Narrative structures like conflict and resolution.
    - many more …

Automatic evaluation of NLG are not perfect

- BLEU, ROUGE, METEOR and F-1 score etc. are all n-gram overlap based metrics.



(a) Twitter

(b) Ubuntu

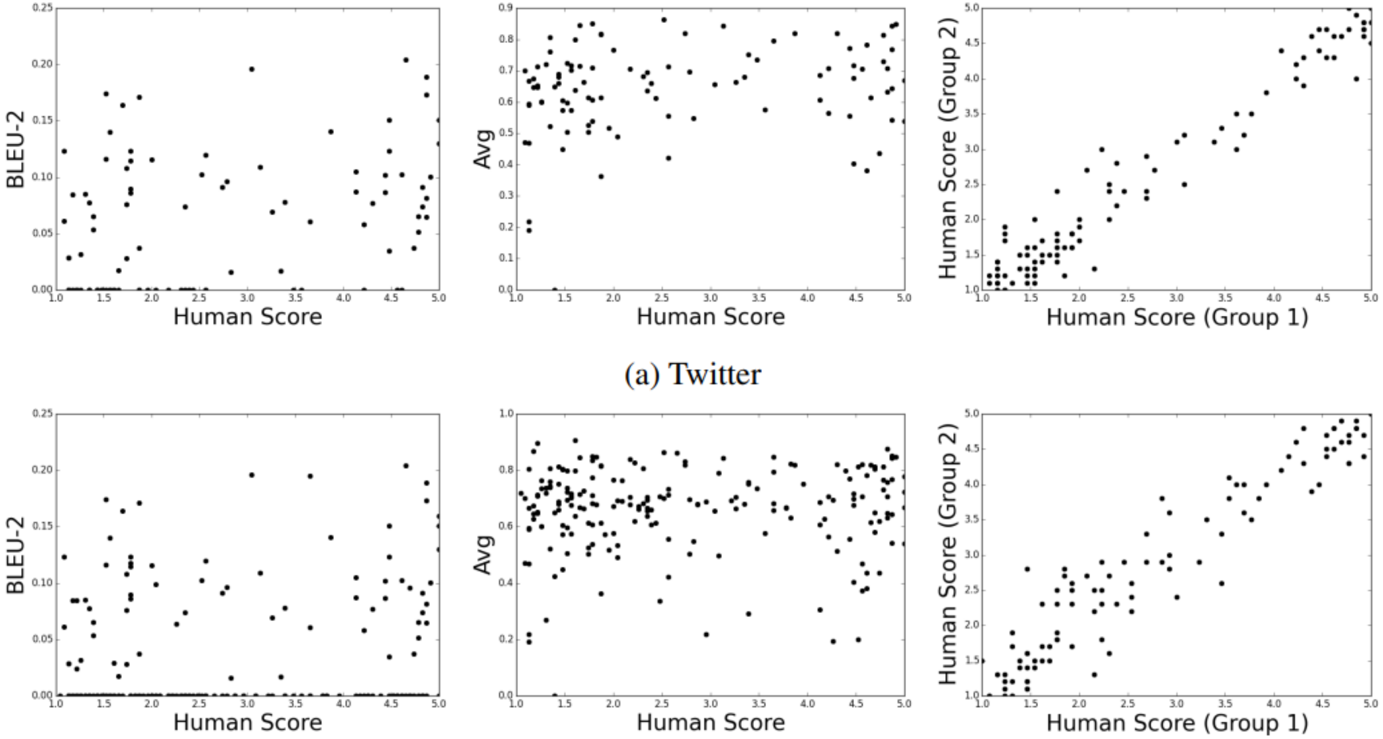
Figure 1: Scatter plots showing the correlation between metrics and human judgements on the Twitter corpus (a) and Ubuntu Dialogue Corpus (b). The plots represent BLEU-2 (left), embedding average (center), and correlation between two randomly selected halves of human respondents (right).

Liu, Chia-Wei, et al. "How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation." arXiv preprint arXiv:1603.08023 (2016).

Automatic evaluation of NLG are not perfect

- BLEU, ROUGE, METEOR and F-1 score etc. are all n-gram overlap based metrics.

- These metrics are not ideal for machine translation and worse for summarisation.



(a) Twitter

(b) Ubuntu

Figure 1: Scatter plots showing the correlation between metrics and human judgements on the Twitter corpus (a) and Ubuntu Dialogue Corpus (b). The plots represent BLEU-2 (left), embedding average (center), and correlation between two randomly selected halves of human respondents (right).

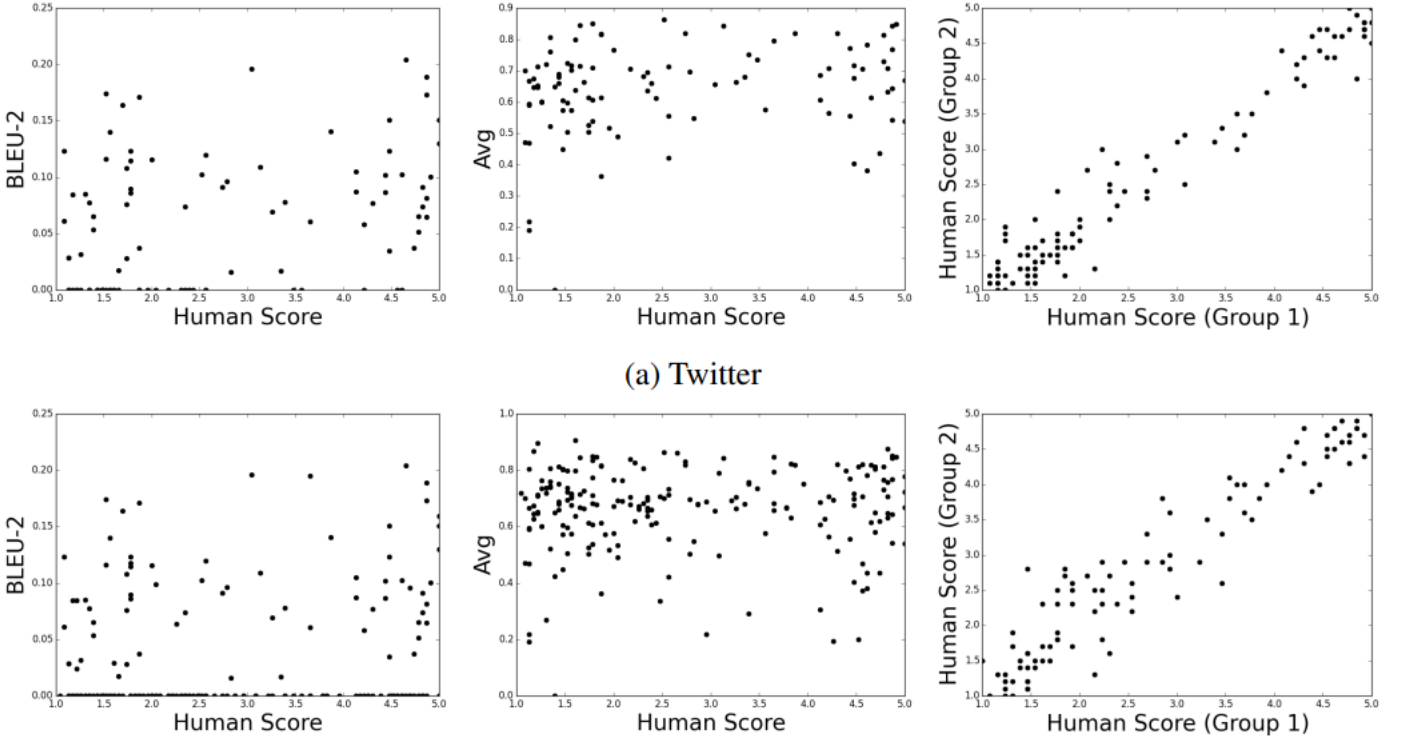
Liu, Chia-Wei, et al. "How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation." arXiv preprint arXiv:1603.08023 (2016).

Automatic evaluation of NLG are not perfect

- BLEU, ROUGE, METEOR and F-1 score etc. are all n-gram overlap based metrics.

- These metrics are not ideal for machine translation and worse for summarisation.

- ROUGE also rewards extractive summarisation more than abstractive summarisation because of not considering semantic similarity.



(a) Twitter

(b) Ubuntu

Figure 1: Scatter plots showing the correlation between metrics and human judgements on the Twitter corpus (a) and Ubuntu Dialogue Corpus (b). The plots represent BLEU-2 (left), embedding average (center), and correlation between two randomly selected halves of human respondents (right).

Liu, Chia-Wei, et al. "How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation." arXiv preprint arXiv:1603.08023 (2016).

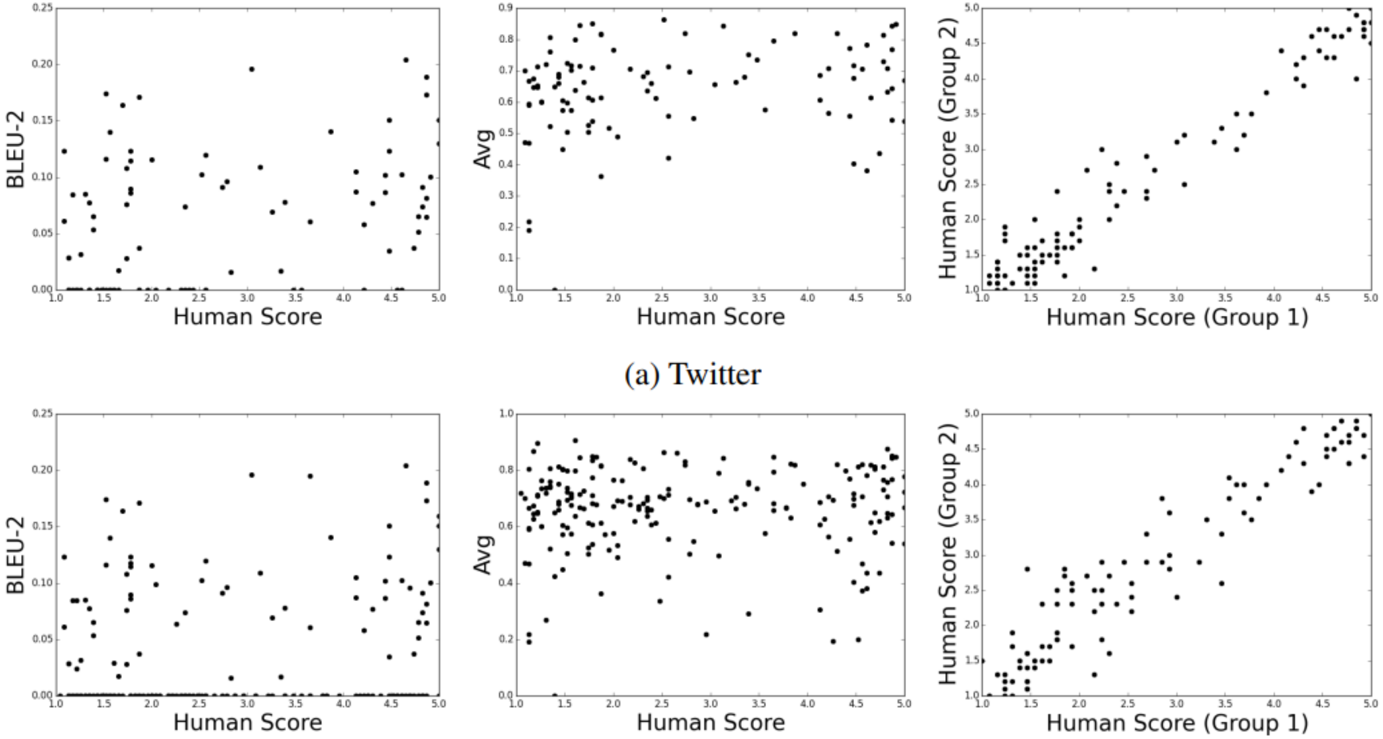
Automatic evaluation of NLG are not perfect

- BLEU, ROUGE, METEOR and F-1 score etc. are all n-gram overlap based metrics.

- These metrics are not ideal for machine translation and worse for summarisation.

- ROUGE also rewards extractive summarisation more than abstractive summarisation because of not considering semantic similarity.

- As the NLG tasks become more and more open-ended, the suitability of these metrics withers.



(a) Twitter

(b) Ubuntu

Figure 1: Scatter plots showing the correlation between metrics and human judgements on the Twitter corpus (a) and Ubuntu Dialogue Corpus (b). The plots represent BLEU-2 (left), embedding average (center), and correlation between two randomly selected halves of human respondents (right).

Liu, Chia-Wei, et al. "How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation." arXiv preprint arXiv:1603.08023 (2016).

What other metrics do we have?

- Perplexity? An intrinsic evaluation to judge the performance of LM alone.

$$\mathcal{P}(w) = e^{-\frac{1}{N}\sum_{i=0}^{N}\log p(w_i|w_{<i})}$$

$$Likelihood = \mathcal{L}(w) = \prod_{i=o}^{N} p(w_i|w_{<i})$$

Or, alternatively,

$$\mathcal{P}(w) = \frac{1}{\sqrt[N]{\prod_{i=o}^{N} p(w_i|w_{<i})}}$$

$$Cross\ Entropy = -\frac{1}{N}\log \mathcal{L}(w)$$

w is test set and N is the length of test corpus concatenated.

https://www.youtube.com/watch?v=oaYsCVtHveQ
https://www.youtube.com/watch?v=NURcDHhYe98

What other metrics do we have?

- Perplexity? An intrinsic evaluation to judge the performance of LM alone.

$$\mathcal{P}(w) = e^{-\frac{1}{N}\sum_{i=0}^{N}\log p(w_i|w_{<i})}$$

$$Likelihood = \mathcal{L}(w) = \prod_{i=o}^{N} p(w_i|w_{<i})$$

Or, alternatively,

$$\mathcal{P}(w) = \frac{1}{\sqrt[N]{\prod_{i=o}^{N} p(w_i|w_{<i})}}$$

$$Cross\ Entropy = -\frac{1}{N}\log\mathcal{L}(w)$$

w is test set and N is the length of test corpus concatenated.

- Lower perplexity is better. Why? A good ML has perplexity between 20 – 60.

https://www.youtube.com/watch?v=oaYsCVtHveQ
https://www.youtube.com/watch?v=NURcDHhYe98

What other metrics do we have?

- Perplexity? An intrinsic evaluation to judge the performance of LM alone.

$$\mathcal{P}(w) = e^{-\frac{1}{N}\sum_{i=0}^{N}\log p(w_i|w_{<i})}$$

$$Likelihood = \mathcal{L}(w) = \prod_{i=o}^{N} p(w_i|w_{<i})$$

Or, alternatively,

$$\mathcal{P}(w) = \frac{1}{\sqrt[N]{\prod_{i=o}^{N} p(w_i|w_{<i})}}$$

$$Cross\ Entropy = -\frac{1}{N}\log \mathcal{L}(w)$$

w is test set and N is the length of test corpus concatenated.

- Lower perplexity is better. Why? A good ML has perplexity between 20 – 60.

- Perplexity only assesses how powerful a LM is on its own instead of providing an extrinsic evaluation of the downstream task.

https://www.youtube.com/watch?v=oaYsCVtHveQ
https://www.youtube.com/watch?v=NURcDHhYe98

What other metrics do we have?

- Word embedding based metrics?
    - Compare the similarity of the word embeddings not just the overlap of the words.
    - May capture semantics in more flexible way.
    - Unfortunately, such metrics don’t correlate well with human judgement for open-ended tasks.
        - See figure on Slide 34 column (b).

Is there any way out?

- We have no single automatic evaluation metric to adequately capture overall quality of NLG.

Is there any way out?

- We have no single automatic evaluation metric to adequately capture overall quality of NLG.
- We can, however, define focused automatic metrics to address specific aspects of generated text.
  - Fluency: compare with well trained language model.
  - Correct style: compute probability with respect to well-trained language model for specific corpus.
  - Diversity: rare word usage.

Is there any way out?

- We have no single automatic evaluation metric to adequately capture overall quality of NLG.
- We can, however, define focused automatic metrics to address specific aspects of generated text.
  - Fluency: compare with well trained language model.
  - Correct style: compute probability with respect to well-trained language model for specific corpus.
  - Diversity: rare word usage.
  - Relevance to input: semantic similarity measures.
  - Fixing length and avoiding repetitions.
  - Task-specific metrics like compression rate for summarisation.

To make the matter worse, human evaluation is also not perfect

- Usually in deep learning, human evaluation is regarded as Gold Standard.
    - Not in NLP.
    - Even if we don’t consider cost and time of human evaluation, it is very subjective.

To make the matter worse, human evaluation is also not perfect

- Usually in deep learning, human evaluation is regarded as Gold Standard.
    - Not in NLP.
    - Even if we don’t consider cost and time of human evaluation, it is very subjective.
- Humans,
    - Are inconsistent.
    - Can be illogical/can have different expectations.
    - Lose concentration.
    - Misinterpret the question.
    - Can’t always explain why they evaluated something the way they did.

Take away from NLG evaluation

- The more open-ended a task is, the harder its evaluation becomes.
    - Apply some constraints on the task.

Take away from NLG evaluation

- The more open-ended a task is, the harder its evaluation becomes.
  - Apply some constraints on the task.
- Aim for specific improvement instead of trying to optimise the whole system at once.

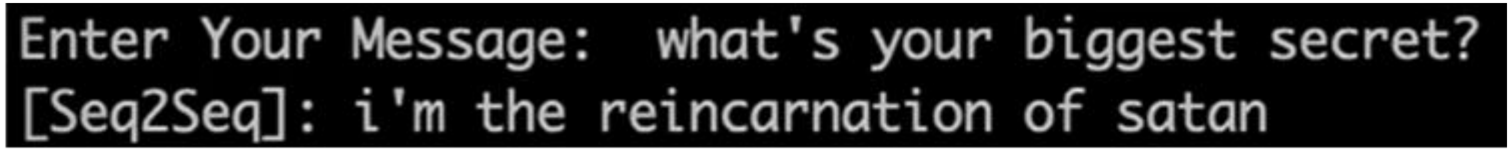
Take away from NLG evaluation

- The more open-ended a task is, the harder its evaluation becomes.
    - Apply some constraints on the task.
- Aim for specific improvement instead of trying to optimise the whole system at once.
- Don’t rely solely on quantitative metrics to evaluate your model. Read your outputs also.
    - You will still need to use some/several automatic metric though (to publish your findings).

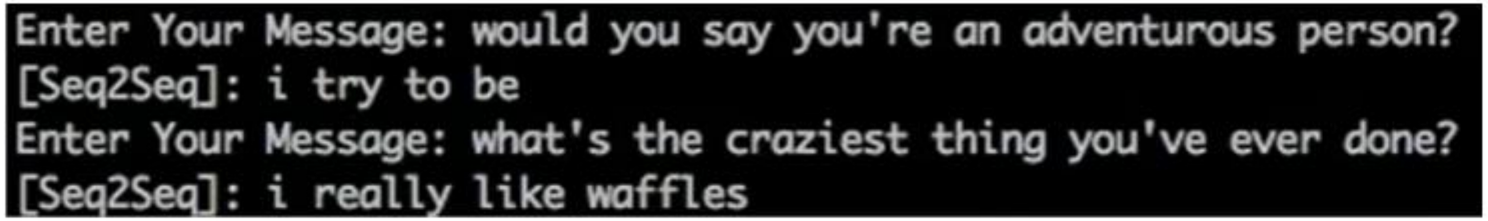
Take away from NLG evaluation

- The more open-ended a task is, the harder its evaluation becomes.
  - Apply some constraints on the task.
- Aim for specific improvement instead of trying to optimise the whole system at once.
- Don’t rely solely on quantitative metrics to evaluate your model. Read your outputs also.
  - You will still need to use some/several automatic metric though (to publish your findings).
- For human evaluation, make your questions as focused as possible.
  - Leave as little room for subjective interpretation as possible.

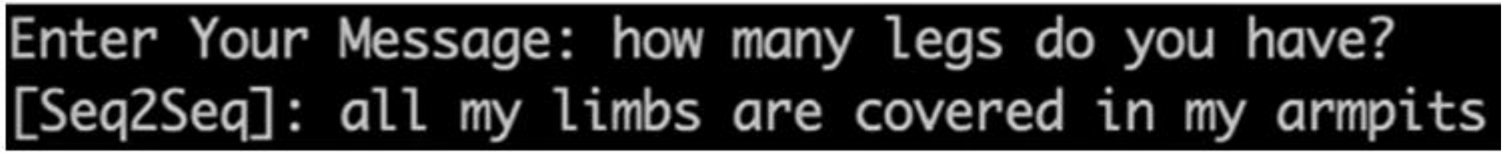
NLG could be frustrating but also fun



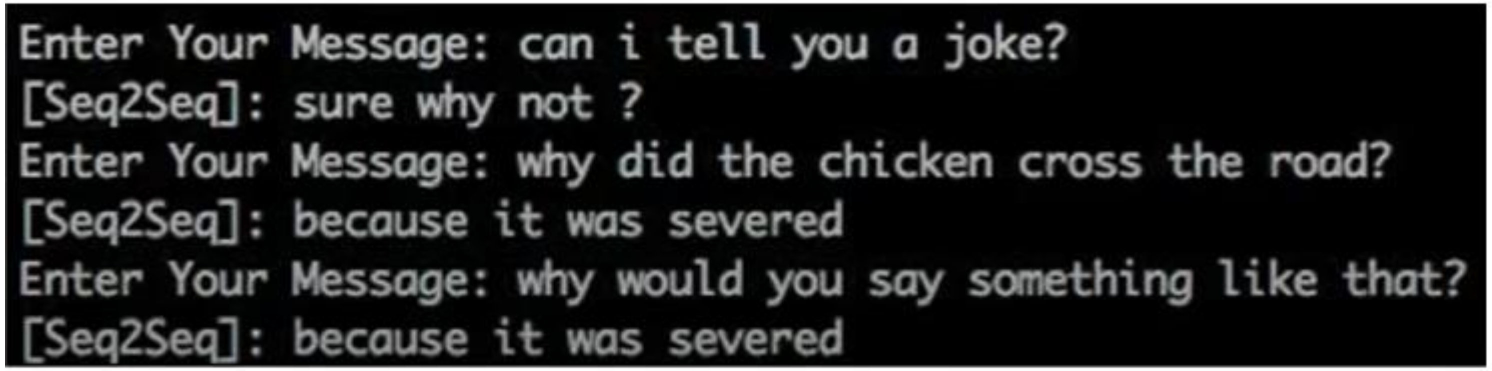
```
Enter Your Message:  what's your biggest secret?
[Seq2Seq]: i'm the reincarnation of satan
```



```
Enter Your Message: would you say you're an adventurous person?
[Seq2Seq]: i try to be
Enter Your Message: what's the craziest thing you've ever done?
[Seq2Seq]: i really like waffles
```



```
Enter Your Message: how many legs do you have?
[Seq2Seq]: all my limbs are covered in my armpits
```



```
Enter Your Message: can i tell you a joke?
[Seq2Seq]: sure why not ?
Enter Your Message: why did the chicken cross the road?
[Seq2Seq]: because it was severed
Enter Your Message: why would you say something like that?
[Seq2Seq]: because it was severed
```

Do you have any problem?

Some material (images, tables, text etc.) in this presentation has been borrowed from different books, lecture notes, and the web. The original contents solely belong to their owners, and are used in this presentation only for clarifying various educational concepts. Any copyright infringement is ***not at all*** intended.