

Generative AI: Decoding and Adaptation

Pierre Colombo

pierre.colombo@centralesupelec.fr

MICS - CentraleSupélec

Advanced Natural Language Processing



Lectures Outline

- 1. The Basics of Natural Language Processing**
- 2. Representing Text with Vectors**
- 3. Deep Learning Methods for NLP**
- 4. Language Modeling / Revision of Transformers**
- 5. Generative AI for NLP / Revision of Transformers**
- 6. Introduction to RAG / Introduction to Distillation**
- 7. Badass Language Modeling: CroissantLLM / TowerLLM**

Labs Outline

1. Describe Statistically large scale corpora
2. Statistical Based and Word2vec Based Retriever
3. Task-Specific Modelling with Neural Networks
4. Task-Specific Modelling with Neural Networks (II)
5. Machine Translation
6. Paper Presentations (4-5-6-7)
7. Paper Presentations (8-9-10)

Today Lecture Outline

- What is NLG
- Decoding Algorithm for NLG
- Evaluating NLG Systems

What is NLG?

What is natural language generation?

Natural language generation is one side of natural language processing. NLP =

Natural Language Understanding (NLU) +
Natural Language Generation (NLG)

NLG focuses on systems that produce **fluent**, **coherent** and **useful** language output for human consumption

Deep Learning is powering next- gen NLG systems!



Example Uses of Natural Language Generation

Machine Translation systems

input: utterances in source languages

output: translated text in target languages.



Digital assistant (dialogue) systems use NLG

input: dialog history

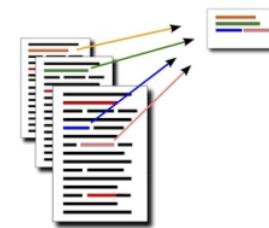
output: text that respond / continue the conversation



Summarization systems (for research articles, email, meetings, documents) use NLG

input: long documents

output: summarization of the long documents

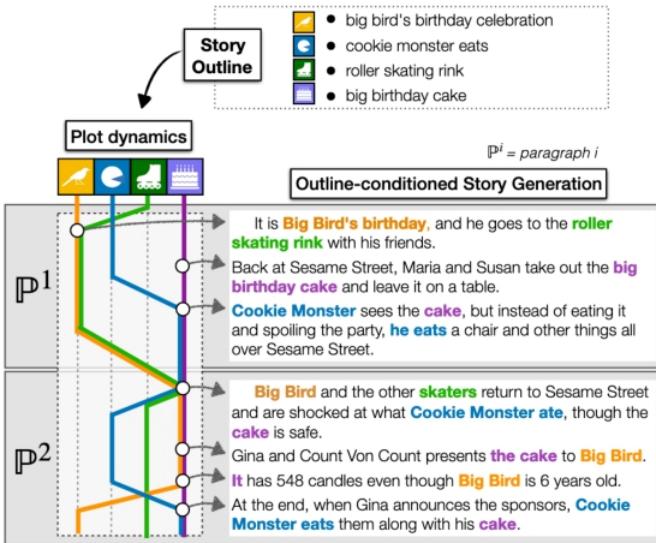


C: Looking at what we've got, we want an LCD display with a spinning wheel.
B: You have to have some push-buttons, don't you?
C: Just spinning and not scrolling, I would say.
B: I think the spinning wheel is definitely very now.
A: but since LCDs seems to be uh a definite yes,
C: We're having push-buttons on the outside,
C: and then on the inside an LCD with spinning wheel,

Decision Abstract (Summary):
The remote will have push buttons outside, and an LCD and spinning wheel inside.

More interesting NLG uses

Creative stories



(Rashkin et al., EMNLP 2020)

Data- to- text

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

YEAR	TEAM	ATT	RUSHING				RECEIVING				
			YDS	AVG	LNG	TD	NO.	YDS	AVG	LNG	
1983	SF	176	725	4.1	71	8	48	427	8.9	23	
1984	SF	155	649	4.2	28	4	71	675	9.5	64	
1985	SF	143	1050	4.9	52	9	92	1105	11.7	6	
1986	SF	204	830	4.1	25	7	81	624	7.7	48	
1987	SF	215	815	3.8	25	3	66	492	7.5	35	
1988	SF	310	1502	4.8	46	9	76	534	7.0	22	
1989	SF	271	1054	3.9	27	6	49	473	9.7	44	
1990	SF	141	439	3.1	26	1	25	201	8.0	31	
1991	RAI	162	590	3.6	15	1	17	136	8.0	20	
1992	MIN	105	416	4.0	21	4	22	164	7.5	22	
1993	MIN	38	119	3.1	11	1	19	169	8.9	31	
Totals		-	1991	8189	4.1	71	56	566	4911	8.7	73

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

(Parikh et al., EMNLP 2020)

Visual description



Two children are sitting at a table in a restaurant. The children are one little girl and one little boy. The little girl is eating a pink frosted donut with white icing lines on top of it. The girl has blonde hair and is wearing a green jacket with a black long sleeve shirt underneath. The little boy is wearing a black zip up jacket and is holding his finger to his lip but is not eating. A metal napkin dispenser is in between them at the table. The wall next to them is white brick. Two adults are on the other side of the short white brick wall. The room has white circular lights on the ceiling and a large window in the front of the restaurant. It is daylight outside.

(Krause et al. CVPR 2017)

SOTA NLG system

ChatGPT is an NLG system!

It's general purpose and can do many NLG tasks!

e.g., Chatbot:

The screenshot shows a conversational interface. A user message at the top asks for birthday ideas. Below it, a response from the bot provides three creative ideas. The interface includes a small profile icon for the bot and a thumbs-up/thumbs-down feedback icon.

XL Got any creative ideas for a 10 year old's birthday?

Here are some creative birthday ideas for a 10 year old:

1. Scavenger Hunt: A birthday scavenger hunt can be a fun and interactive activity for kids. You can create a series of clues that lead the birthday child to the next one, until they reach the final prize.
2. DIY Crafts: Crafts are always a hit with kids. You can arrange a crafting session where the birthday child and their friends can make their own personalized gifts or decorations.
3. Movie Night: Have a movie night with friends and family at home or at a local theater. You can also make it a sleepover by setting up sleeping bags, pillows, and blankets.

SOTA NLG system

ChatGPT is an NLG system!
It's general purpose and can do many NLG tasks!

e.g., Poetry Generation:

XL

Write me a poem about sorting algorithms



Sorting algorithms, oh so wise,
Bringing order to chaotic ties.
Bubble sort, quick sort, merge sort too,
Each one unique, with their own special view.

Bubble sort, gentle as can be,
Compares and swaps, so efficiently.
It moves like waves, across the sea,
Bringing order, so beautifully.

Quick sort, so fast and strong,
Divides and conquers all day long.
With each partition, it grows so wide,
Order is found, on every side.

Merge sort, so neat and trim,
Divides and merges, one by one.
It builds a bridge, to conquer all,
Order is its overall call.

SOTA NLG system

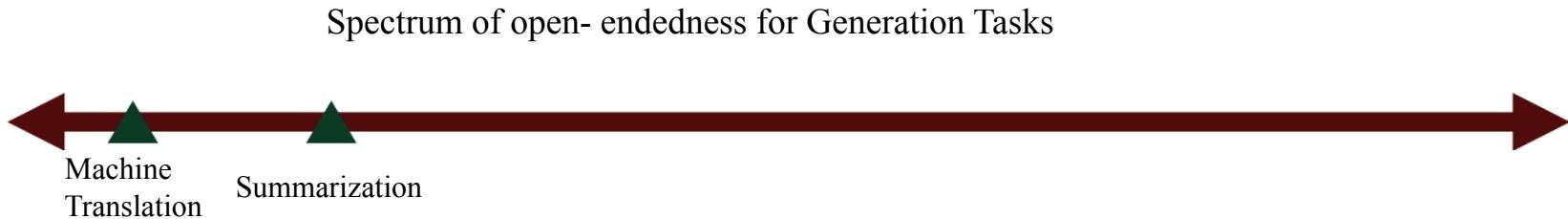
Reinventing search with a new AI-powered Microsoft Bing and Edge, your copilot for the web

The screenshot shows the Microsoft Bing search interface. At the top, there is a navigation bar with the Microsoft Bing logo, a search bar labeled "SEARCH", and a "CHAT" button which is underlined, indicating it is active. To the right of the search bar are icons for notifications and a menu. Below the navigation bar, a blue message box contains a user query: "I am planning a trip for our anniversary in September. What are some places we can go that are within a 3 hour flight from London Heathrow?". A response message below it says: "Congratulations on your anniversary! 🎉 There are many places you can go that are within a 3 hour flight from London Heathrow. Here are some suggestions based on your preferences and the best destinations in Europe in September [4] [5] [6]:". Three bullet points follow, listing travel destinations:

- If you like **beaches** and **sunshine**, you can fly to **Malaga** in Spain ^[1], which has a pleasant climate, a lively culture, and a beautiful coastline. You can enjoy the sandy beaches, the historic monuments, and the delicious tapas. Malaga is also a great base to explore other nearby attractions, such as Granada, Seville, and Cordoba.
- If you like **mountains** and **lakes**, you can fly to **Annecy** in France ^[4], which is a charming alpine town near the Swiss border. Annecy is known as the "Venice of France" because of its canals and bridges. You can admire the stunning scenery, visit the medieval castle, and try some local cheese and wine. Annecy is also close to Geneva, Chamonix, and Lyon, if you want to see more of the region.
- If you like **art** and **history**, you can fly to **Florence** in Italy ^[6], which is the birthplace of the Renaissance and a UNESCO World Heritage Site. Florence is a treasure trove of artistic and architectural masterpieces, such as the Duomo, the Uffizi Gallery, and the Ponte Vecchio. You can also explore the Tuscan countryside, taste the famous gelato, and shop for leather goods.

At the bottom of the interface, there is a blue button with a microphone icon labeled "Ask me anything..." and a decorative landscape image at the very bottom.

Categorization of NLG tasks



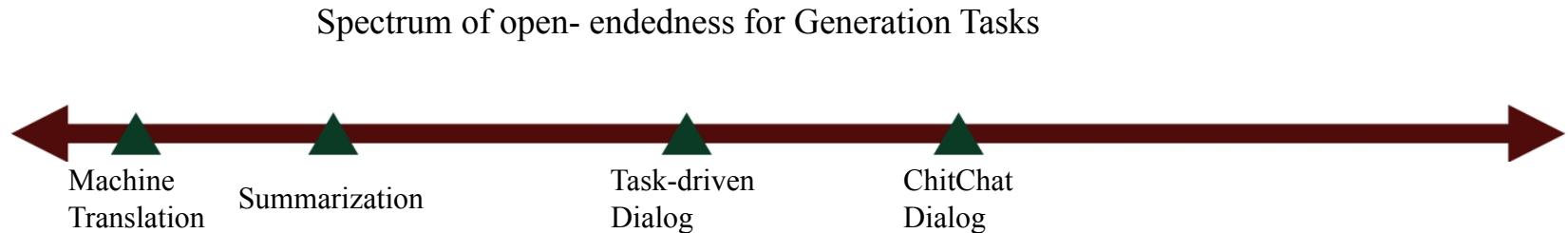
Source Sentence: 当局已经宣布今天是节假日。

Reference Translation:

1. Authorities have announced a national holiday today.
2. Authorities have announced that today is a national holiday.
3. Today is a national holiday, announced by the authorities.

The output space is not very diverse.

Categorization of NLG tasks



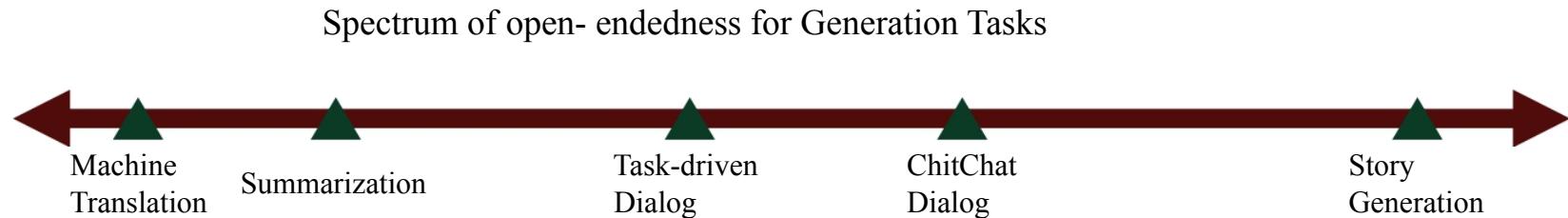
Input: Hey, how are you?

Outputs:

1. Good! You?
2. I just heard an exciting news, do you want to hear it?
3. Thx for asking! Barely surviving my hws.

The output space is getting more diverse...

Categorization of NLG tasks



Input: Write a story about three little pigs?

Outputs:

... (so many options) ...

The output space is extremely diverse...

Categorization of NLG tasks

Less Open -ended

More Open -ended



Open - ended generation : the output distribution still has high freedom

Non - open - ended generation : the input mostly determines the output generation.

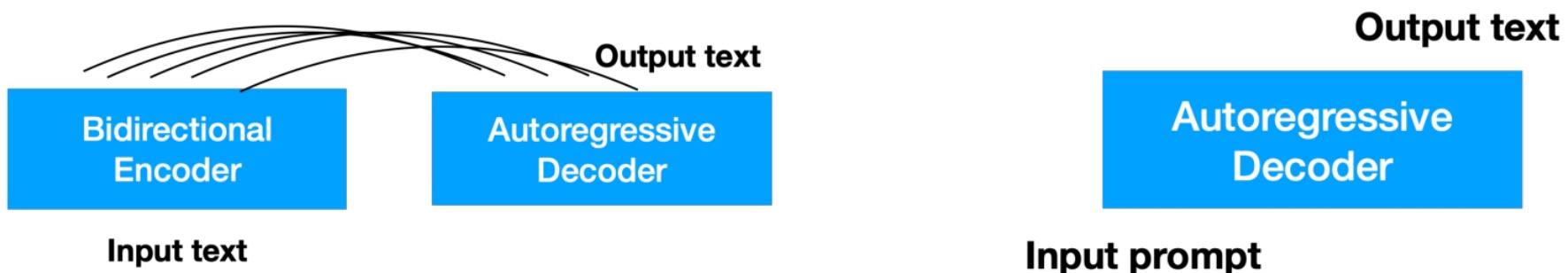
Remark: One way of formalizing categorization this is by **entropy**.

These two classes of NLG tasks require different decoding and/or training approaches!

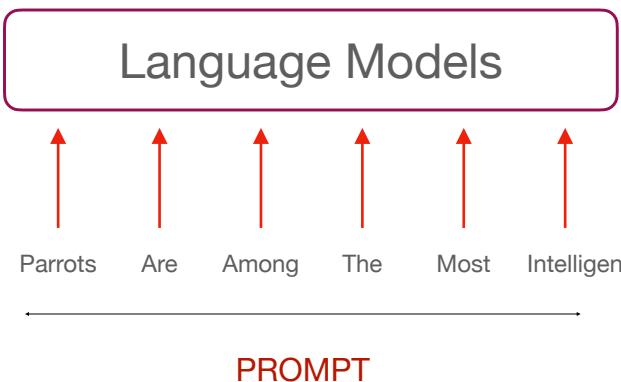
Decoding Algorithms For NLG

Basics of natural language generation

- For non - open - ended tasks (e.g., MT), we typically use a encoder - decoder system, where this autoregressive model serves as the decoder, and we'd have another bidirectional encoder for encoding the inputs.
- For open - ended tasks (e.g., story generation), this autoregressive generation model is often the only component.

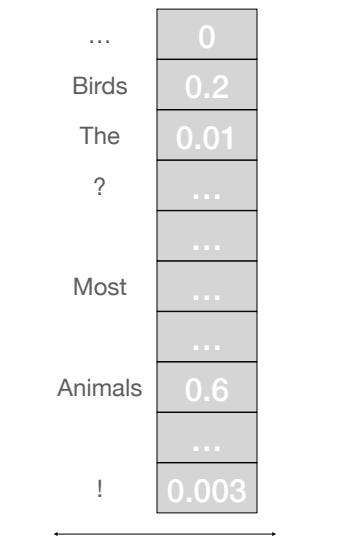


Language Model at a Glance

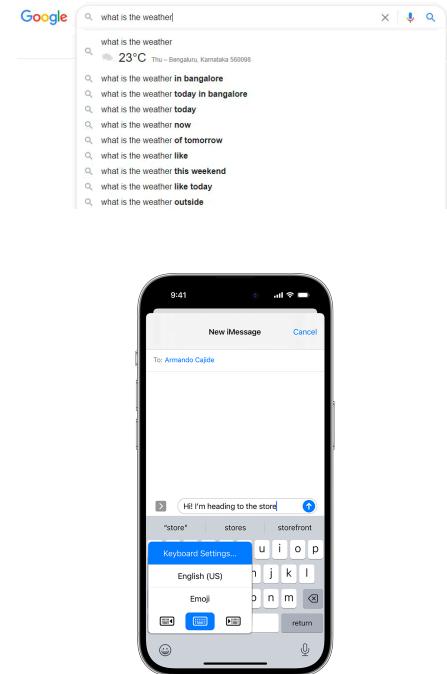


« Initial input or instruction given to a language model to generate a specific response or continuation of text. »

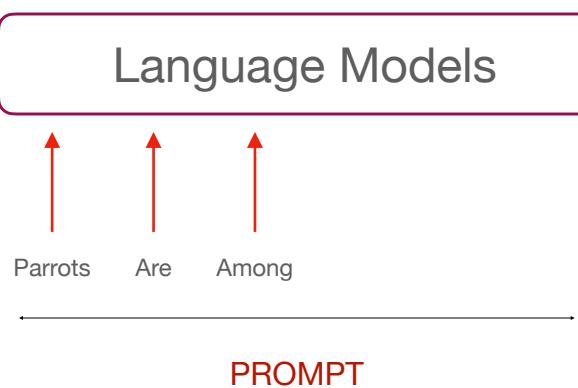
Predict



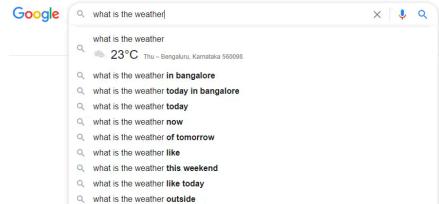
Next word prediction



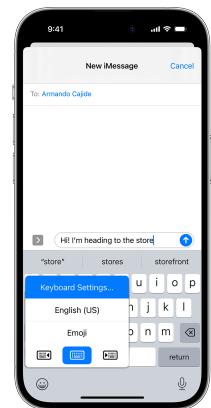
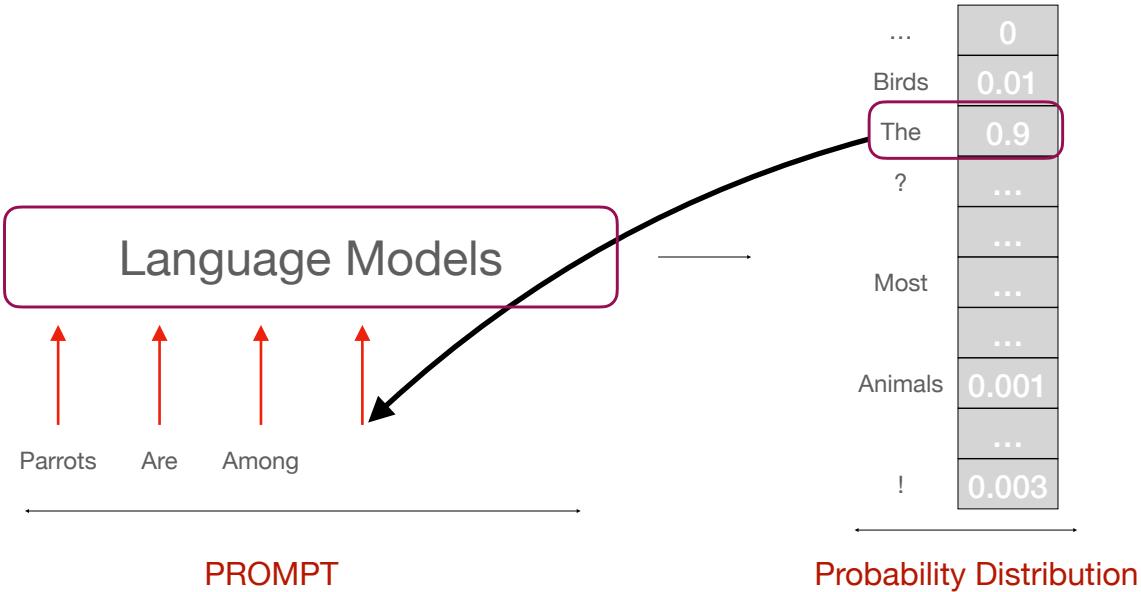
Generating with Language Models



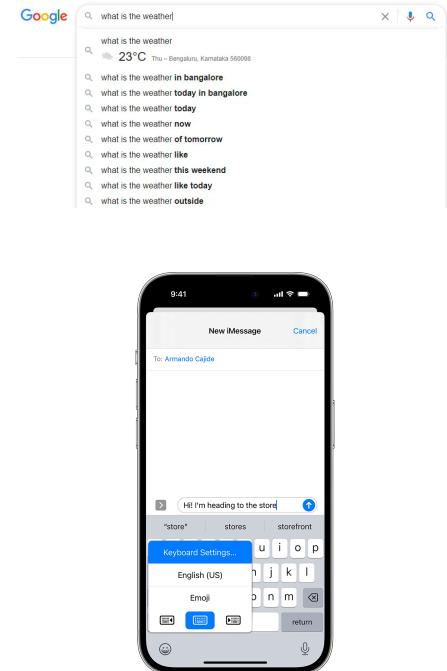
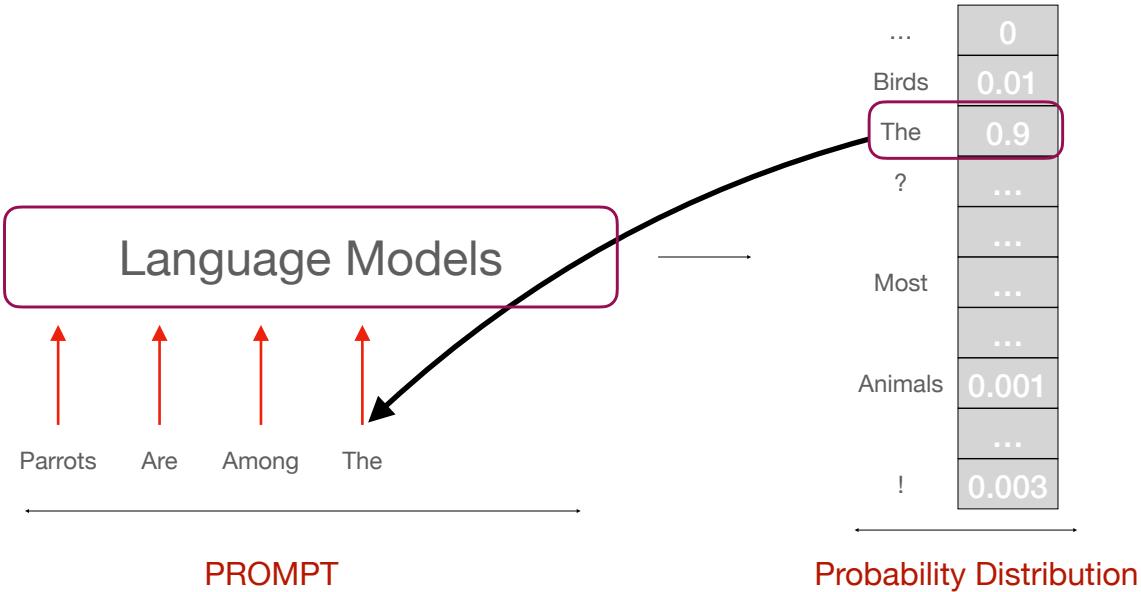
« Initial input or instruction given to a language model to generate a specific response or continuation of text. »



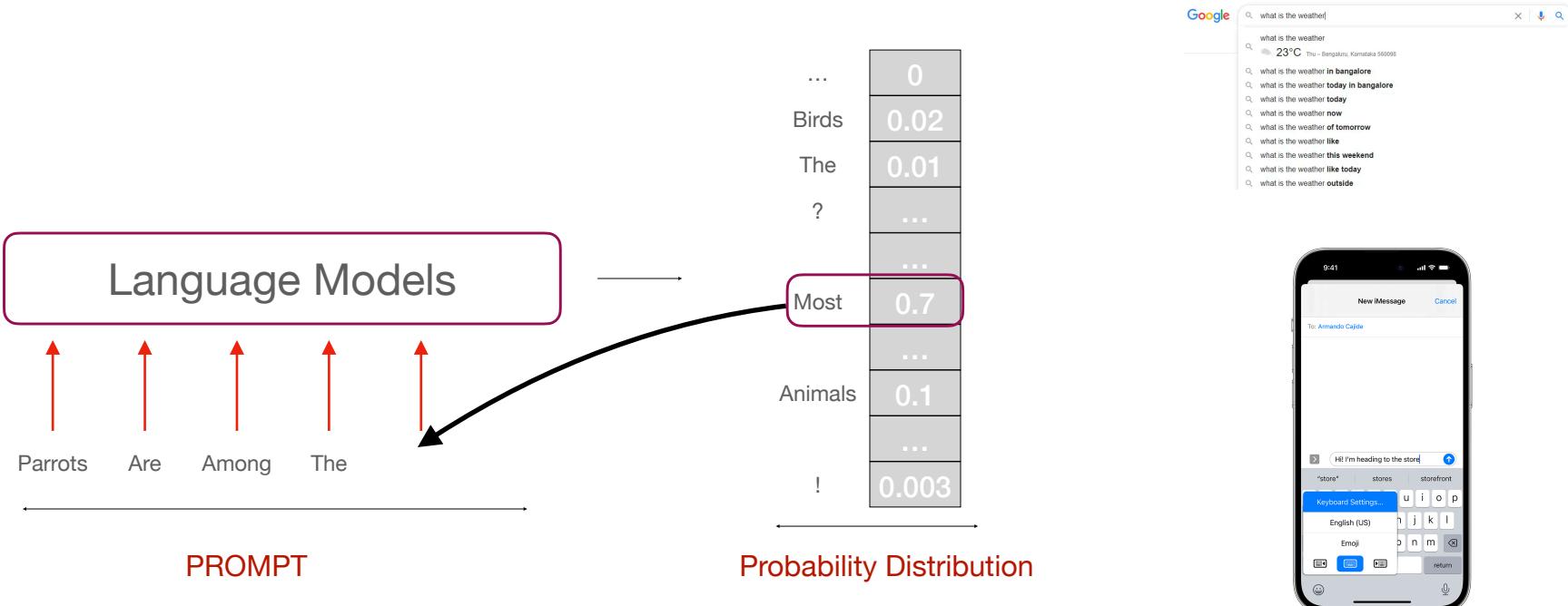
Generating with Language Models



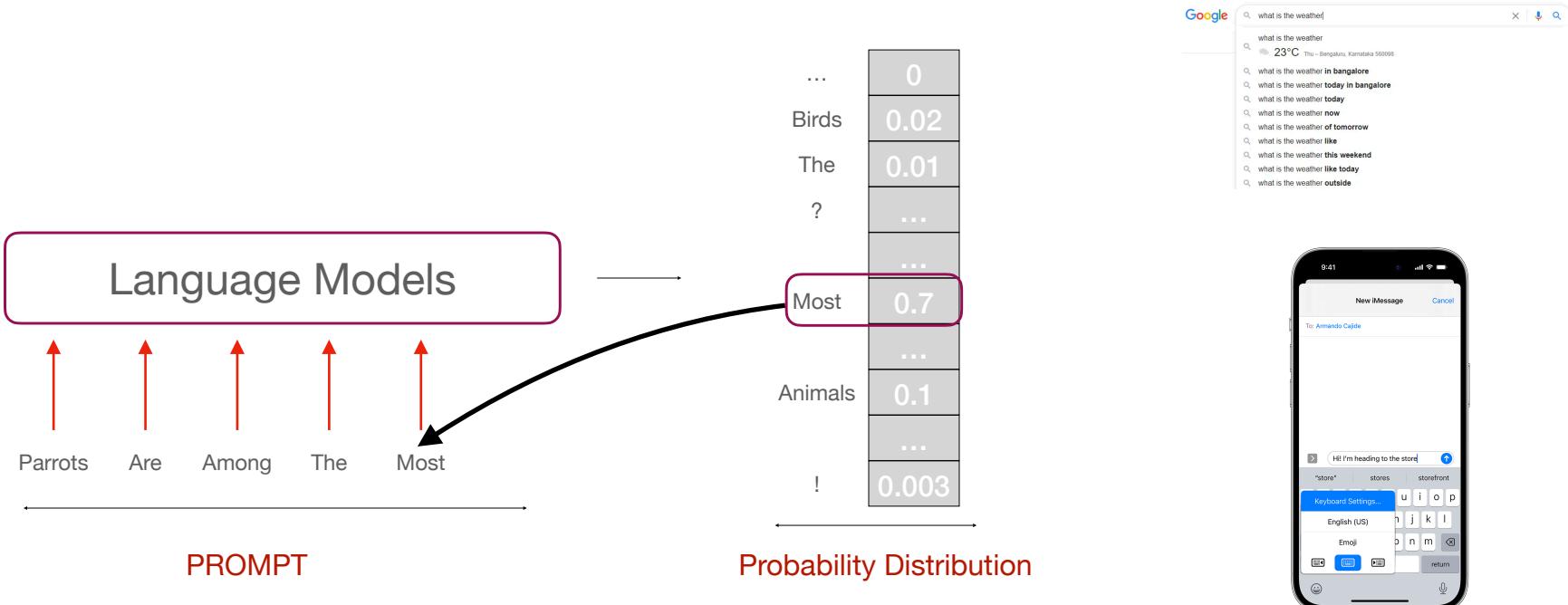
Generating with Language Models



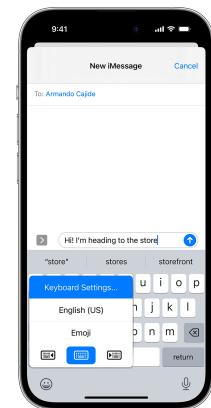
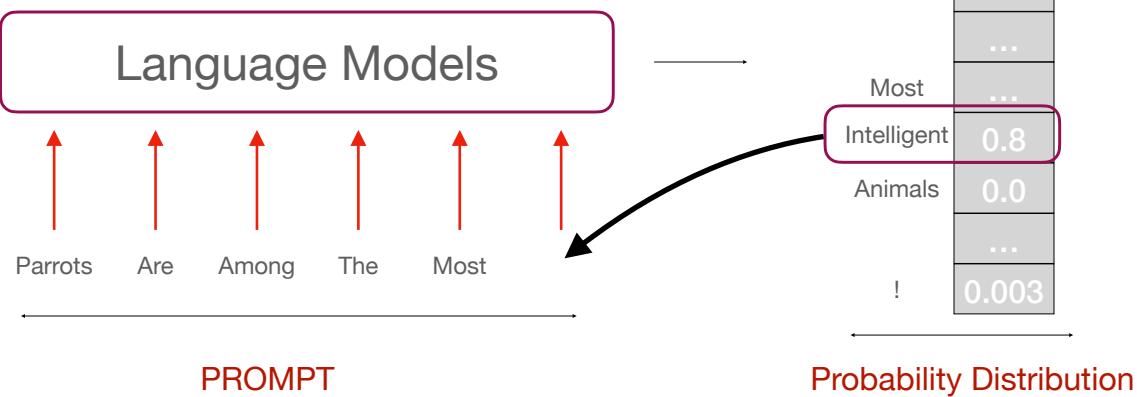
How to use all the data from the web?



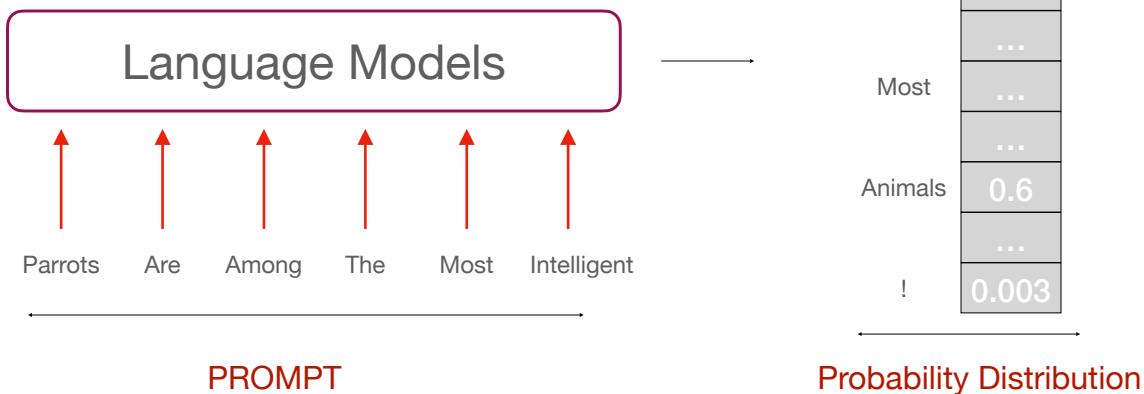
How to use all the data from the web?



How to use all the data from the web?



How to use all the data from the web?



Parrots are among the most intelligent animals

On the Dangers of
Stochastic Parrots:
Can Language Models
Be Too Big?



Emily M. Bender*, Timnit Gebru*,
Angelina McMillan-Major*, Shmargaret Shmatcova,
University of Washington † Back in AI ‡ The ArXiv
*These authors contributed equally.

Basics of natural language generation

- At inference time, our decoding algorithm defines a function to select a token from this distribution:

$$\hat{y}_t = g(P(y_t | y_{\leq t}))$$

`g(.)` is your decoding algorithm

- The “obvious” decoding algorithm is to greedily choose the highest probability next token according to the model at each time step
- While this basic algorithm sort of works, to do better, the two main avenues are to:
 1. Improve decoding
 2. Improve the training

Of course, there's also improving your training data or model architecture

Decoding: what is it all about?

- At each time step t , our model computes a vector of scores for each token in our vocabulary, $S \in R^V$

$$S = f(y_{\leq t})$$

$f(\cdot)$ Is your model

- Then, we compute a probability distribution \cdot over these scores with a softmax function:

$$P(y_t = w | y_{\leq t}) = \frac{\exp(S_w)}{\sum_{v \in V} \exp(S_v)}$$

- Our decoding algorithm defines a function to select a token from this distribution:

$$\hat{y}_t = g(P(y_t | y_{\leq t}))$$

$g(\cdot)$ Is the decoding algorithm

How to find the most likely string?

- Recall: On Neural Machine Translation...
- **Greedy Decoding**
 - Selects the highest probability token

$$\hat{y}_t = \operatorname{argmax}(P(\cdot | y_{\leq t}))$$

- **Beam Search**
 - Also aims to find strings that maximize the log- prob, but with wider exploration of candidates

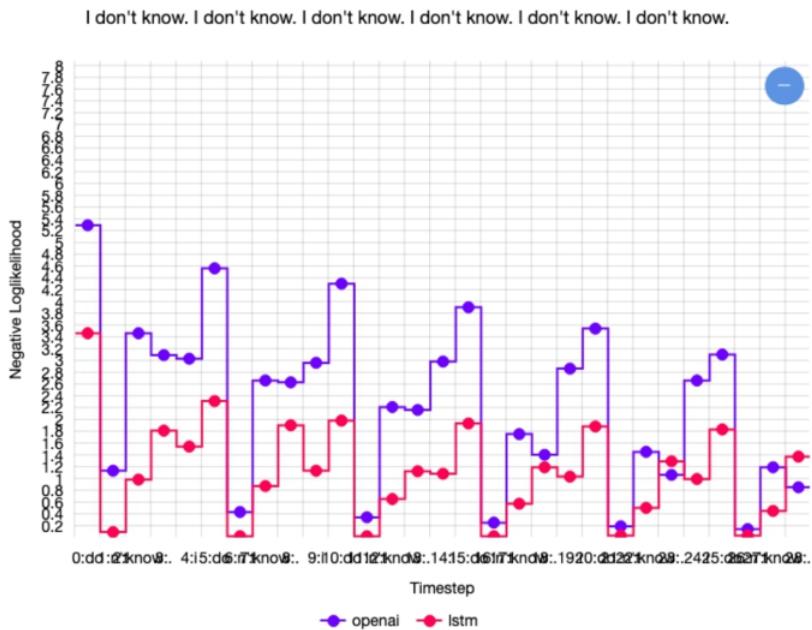
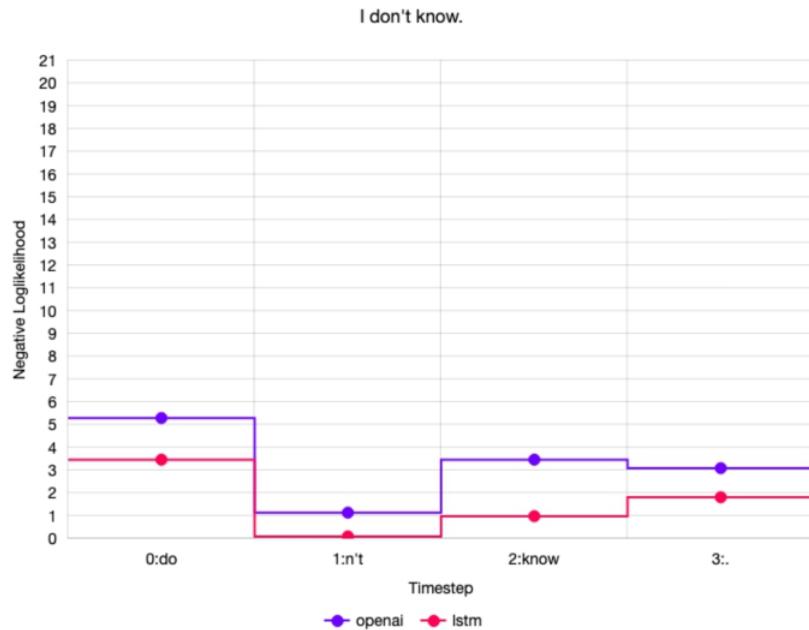
Overall, maximum probability decoding is good for low- entropy tasks like MT and summarization!

The most likely string is repetitive for Open -ended Generation

Context: In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.

Continuation: The study, published in the Proceedings of the National Academy of Sciences of the United States of America (PNAS), was conducted by researchers from the **Universidad Nacional Autónoma de México (UNAM)** and **the Universidad Nacional Autónoma de México (UNAM/Universidad Nacional Autónoma de México/Universidad Nacional Autónoma de México/Universidad Nacional Autónoma de México/Universidad Nacional Autónoma de México...)**

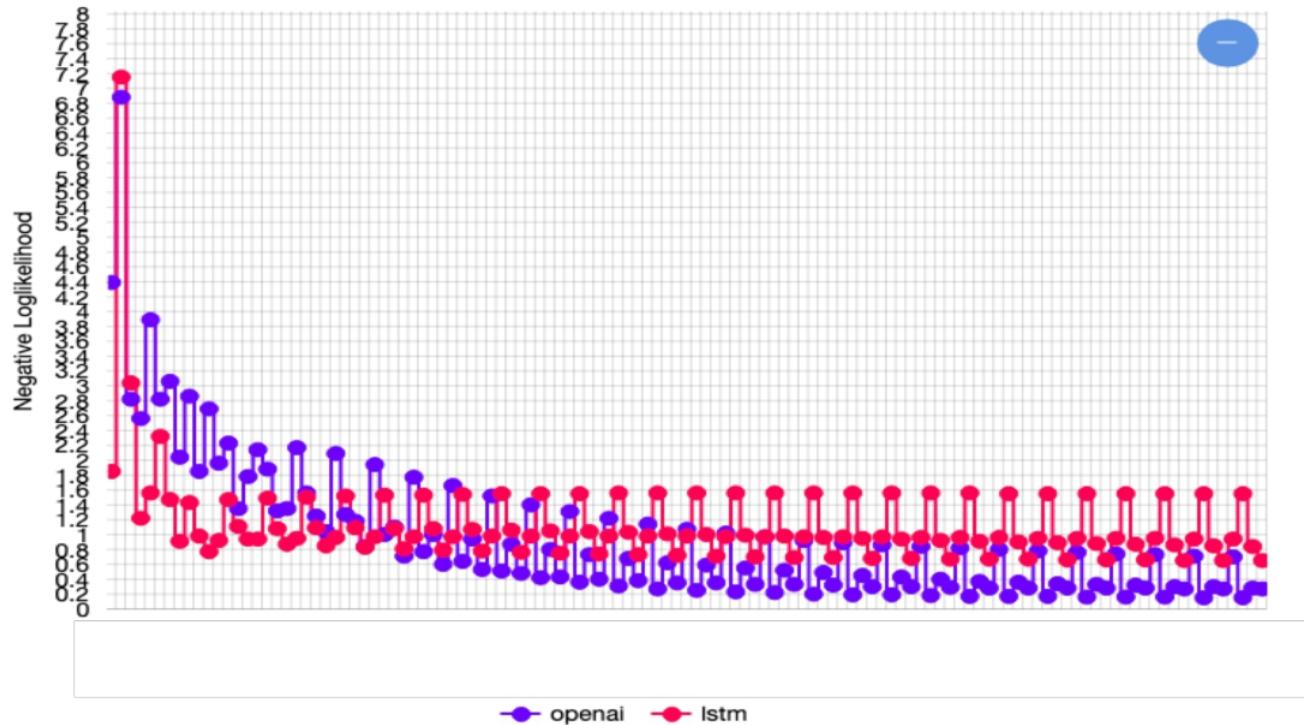
Why does repetition happen?



A self- amplification effect!

And it keeps going...

I'm tired. I'm tired.



Scale doesn't solve this problem: even a 175 billion parameter LM still repeats when we decode for the most likely string.

(Holtzman et. al., ICLR 2020)

How can we reduce repetition?

Simple option:

- Heuristic: Don't repeat n -grams

More complex:

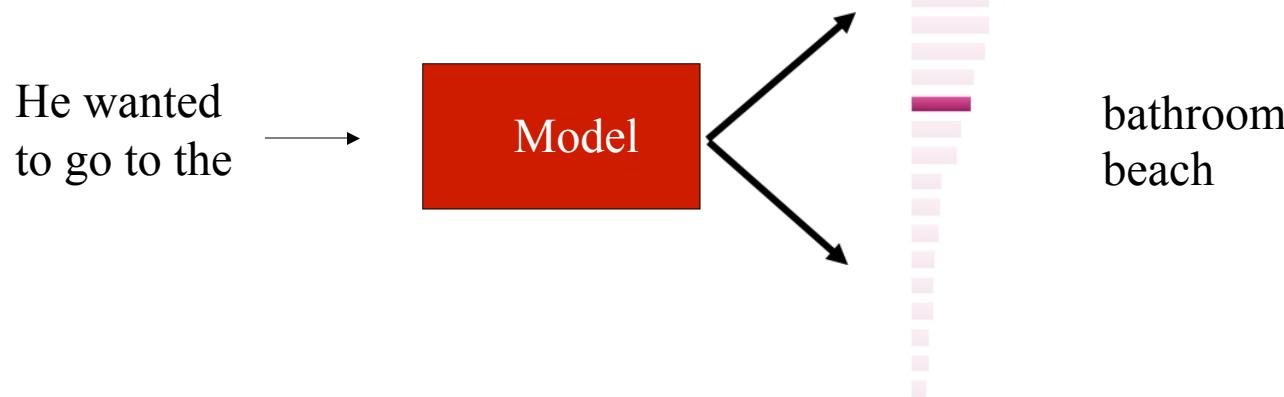
- Use a different training objective:
 - Unlikelihood objective (Welleck et al., 2020) penalize generation of already - seen tokens
 - Coverage loss (See et al., 2017) Prevents attention mechanism from attending to the same words
- Use a different decoding objective:
 - Contrastive decoding (Li et al, 2022) searches for strings x that maximize $\text{logprob_largeLM}(x) - \text{logprob_smallLM}(x)$.

Time to get random : Sampling!

- Sample a token from the distribution of tokens

$$\hat{y}_t \sim P(\cdot | y_{\leq t})$$

- It's *random* so you can sample any token!

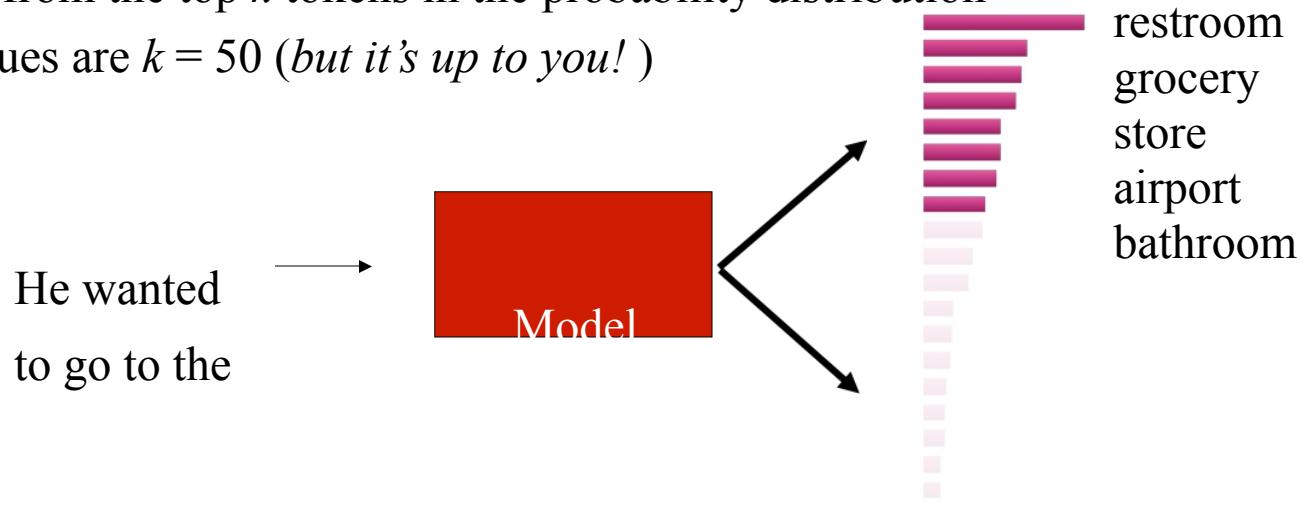


Decoding: Top- k sampling

- Problem: Vanilla sampling makes every token in the vocabulary an option
 - Even if most of the **probability mass** in the distribution is over a limited set of options, the tail of the distribution could be very long and in aggregate have considerable mass (statistics speak: we have “**heavy tailed**” distributions)
 - Many tokens are probably *really wrong* in the current context
 - For these wrong tokens, we give them *individually* a tiny chance to be selected.
 - But because there are many of them, we still give them *as a group* a high chance to be selected.
- Solution: Top - k sampling
 - Only sample from the top k tokens in the probability distribution

Decoding: Top- k sampling

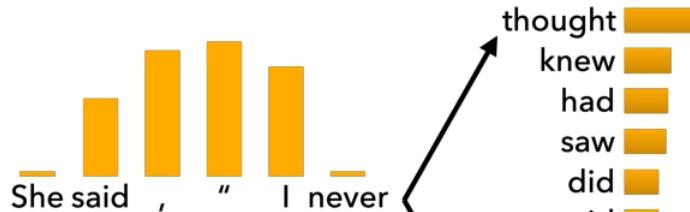
- Solution: Top - k sampling
 - Only sample from the top k tokens in the probability distribution
 - Common values are $k = 50$ (*but it's up to you!*)



- Increase k yields more **diverse**, but **risky** outputs
- Decrease k yields more **safe** but **generic** outputs

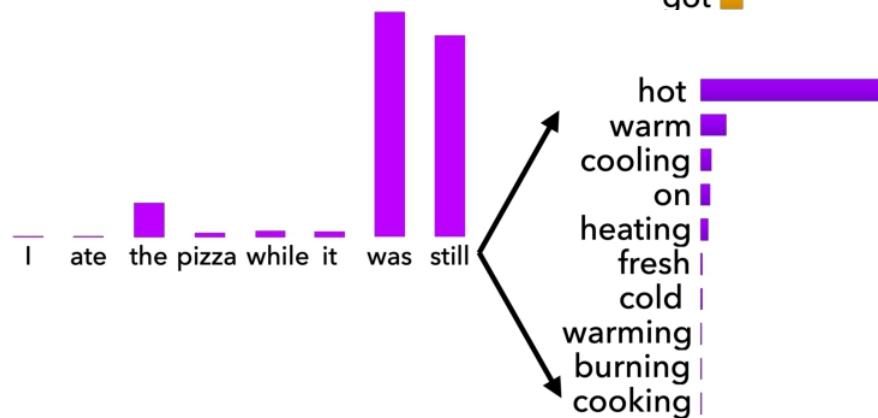
(Fan et al., ACL 2018; Holtzman et al., ACL 2018)

Issues with Top- k sampling



thought
knew
had
saw
did
said
wanted
told
liked
qot

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



hot
warm
cooling
on
heating
fresh
cold
warming
burning
cooking

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

Decoding: Top- p (nucleus) sampling

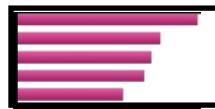
- Problem: The probability distributions we sample from are dynamic
 - When the distribution P_t is flatter, a limited k removes many viable options
 - When the distribution P_t is peakier, a high k allows for too many options to have a chance of being selected
- Solution: Top - p sampling
 - Sample from all tokens in the top p cumulative probability mass (i.e., where mass is concentrated)
 - Varies k depending on the uniformity of P_t

(Holtzman et. al., ICLR 2020)

Decoding: Top- p (nucleus) sampling

- Solution: Top - p sampling
 - Sample from all tokens in the top p cumulative probability mass (i.e., where mass is concentrated)
 - Varies k depending on the uniformity of P_t

$$\hat{y}_1 \sim P(\cdot | y_{\leq 1})$$



$$\hat{y}_2 \sim P(\cdot | y_{\leq 2})$$



$$\hat{y}_3 \sim P(\cdot | y_{\leq 3})$$



Scaling randomness: Temperature

- Recall: On timestep t , the model computes a prob distribution P_t by applying the softmax function to a vector of scores
- You can apply a *temperature hyperparameter* to the softmax to rebalance :

$$P(y_t = w | y_{\leq t}) = \frac{\exp(S_w/\tau)}{\sum_{v \in V} \exp(S_v/\tau)}$$
$$\tau \in R^+ \geq 1$$

- Raise the temperature $\tau \in R^+ \geq 1$ becomes more uniform
 - More diverse output (probability is spread around vocab)
- Lower the temperature $\tau \in R^+ \leq 1$ becomes more spiky
 - Less diverse output (probability is concentrated on top words)

Temperature is a hyperparameter for decoding:
It can be tuned for both beam search and sampling.

Improving Decoding: Re -ranking

- Problem: What if I decode a bad sequence from my model?
- **Decode a bunch of sequences**
 - 10 candidates is a common number, but it's up to you
- Define a score to approximate quality of sequences and **re - rank by this score**
 - Simplest is to use (low) **perplexity** !
 - Careful! Remember that **repetitive utterances** generally get low perplexity.
 - Re - rankers can score a **variety of properties**:
 - style ([Holtzman et al., 2018](#)) , discourse ([Gabriel et al., 2021](#)) , entailment/factuality ([Goyal et al., 2020](#)) , logical consistency ([Lu et al., 2020](#)) , and many more ...
 - Beware poorly- calibrated re - rankers
 - Can compose multiple re - rankers together.

Decoding: Takeaways

- Decoding is still a challenging problem in NLG – there's a lot more work to be done!
- Different decoding algorithms can allow us to inject biases that encourage different properties of coherent natural language generation
- Some of the most impactful advances in NLG of the last few years have come from simple but effective modifications to decoding algorithms

How to evaluate NLG systems

Types of evaluation methods for text generation

Ref: They walked **to the grocery store** .
Gen: **The woman went to the hardware store** .



Content Overlap Metrics



Model- based Metrics



Human Evaluations

Content overlap metrics

Ref: They walked **to the grocery store**.

Gen: **The woman went to the hardware store .**

The diagram illustrates content overlap between two sentences. The reference sentence is "They walked to the grocery store." The generated sentence is "The woman went to the hardware store." Three words are highlighted in blue: "to the grocery store" in the reference and "the hardware store" in the generation. Arrows point from each of these three words to their corresponding words in the other sentence, indicating they are considered matching tokens.

- Compute a score that indicates the lexical similarity between *generated* and *gold-standard (human-written) text*
- Fast and efficient and widely used
- N -gram overlap metrics (e.g., BLEU, ROUGE, METEOR, CIDEr , etc.)

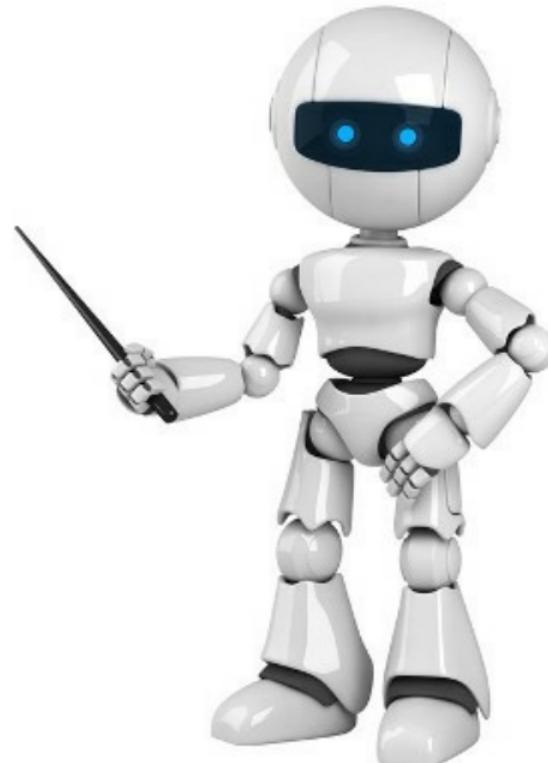
N-gram overlap metrics

Word overlap – based metrics (BLEU, ROUGE, METEOR, CIDEr , etc.)

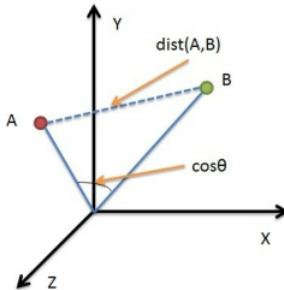
- They're **not ideal for machine translation**
- They get progressively **much worse** for tasks that are more open - ended than machine translation
 - **Worse** for **summarization**, as longer output texts are harder to measure
 - **Much worse** for **dialogue** , which is more open - ended than summarization
 - **Much, much worse** **story generation**, which is also open- ended, but whose sequence length can make it seem you're getting decent scores!

Model -based metrics to capture more semantics

- Use learned representations of words and sentences to compute semantic similarity between generated and reference texts
- No more **n - gram bottleneck** because text units are represented as **embeddings!**
- The embeddings are **pretrained**, distance metrics used to measure the similarity can be **fixed**



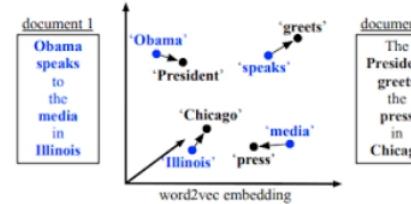
Model -based metrics: Word distance functions



Vector Similarity

Embedding based similarity for semantic distance between text.

- Embedding Average (Liu et al., 2016)
- Vector Extrema (Liu et al., 2016)
- MEANT (Lo, 2017)
- YISI (Lo, 2019)



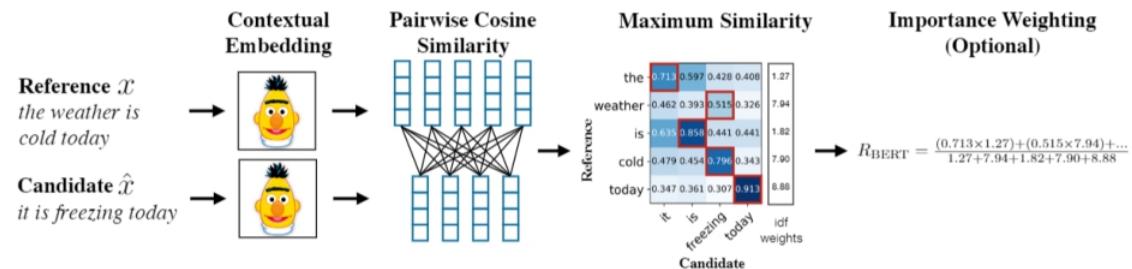
Word Mover's Distance

Measures the distance between two sequences (e.g., sentences, paragraphs, etc.), using word embedding similarity matching.

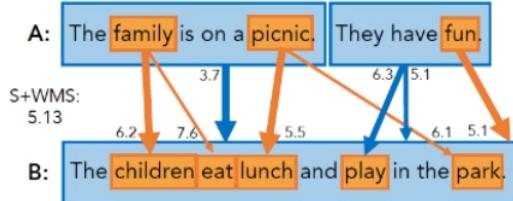
(Kusner et.al., 2015; Zhao et al., 2019)

BERTSCORE

Uses pre-trained contextual embeddings from BERT and matches words in candidate and reference sentences by cosine similarity.
(Zhang et.al. 2020)



Model -based metrics: Beyond word matching



Sentence Movers Similarity :

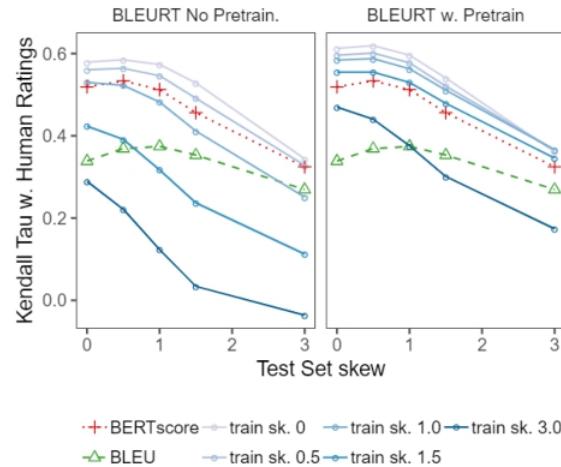
Based on Word Movers Distance to evaluate text in a continuous space using sentence embeddings from recurrent neural network representations.

(Clark et.al., 2019)

BLEURT:

A regression model based on BERT returns a score that indicates to what extent the candidate text is grammatical and conveys the meaning of the reference text.

(Sellam et.al. 2020)



How to evaluate an evaluation metric?

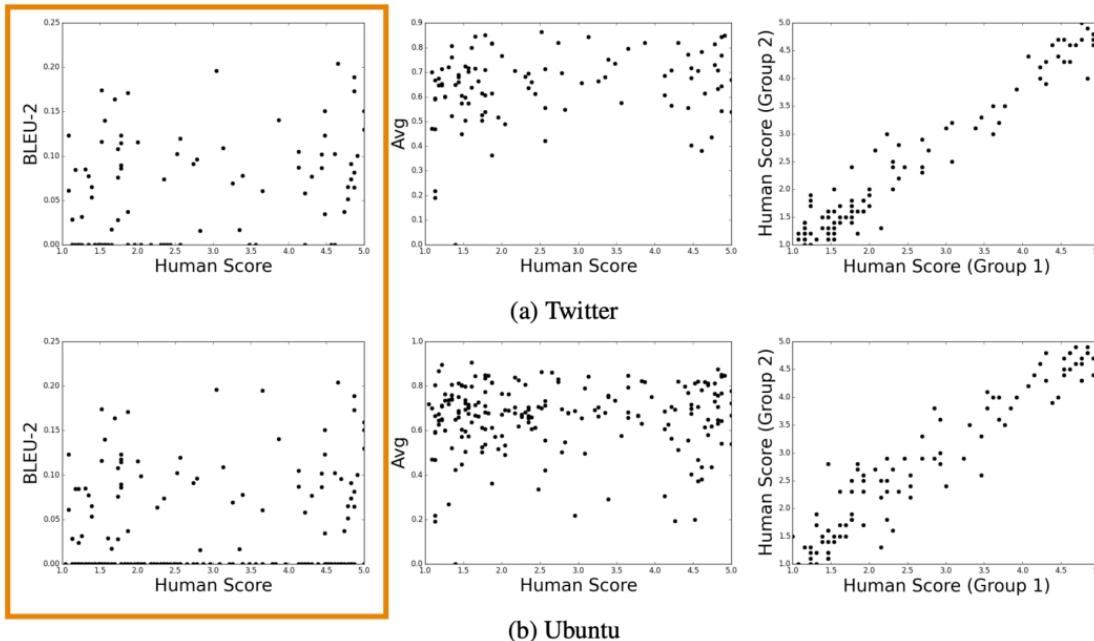


Figure 1: Scatter plots showing the correlation between metrics and human judgements on the Twitter corpus (a) and Ubuntu Dialogue Corpus (b). The plots represent BLEU-2 (left), embedding average (center), and correlation between two randomly selected halves of human respondents (right).

(Liu et al, EMNLP 2016)

Human evaluation: Issues

- Human judgments are regarded as the **gold standard**
- Of course, we know that human eval is **slow** and **expensive**
- Beyond the cost of human eval, it's still far from perfect:
- Humans Evaluation is hard:
 - Results are inconsistent / not reproducible
 - can be illogical
 - misinterpret your question
 - Precision not recall.
 - ...

Evaluation: Takeaways

- *Content overlap metrics* provide a good starting point for evaluating the quality of generated text, but they're **not good enough on their own** .
- *Model - based metrics* can be **more correlated with human judgment**, but behavior is **not interpretable**
- *Human judgments* are critical
 - **But humans are inconsistent!**
- In many cases, the best judge of output quality is **YOU!**
 - **Look at your model generations. Don't just rely on numbers!**
 - **Publicly release large samples of the output of systems that you create!**

Lecture Summary

- **Introduction to NLG**
- **Decoding Algorithm**
- **Some evaluation of NLG**
- **Evaluating Evaluation of NLG**