

Introduction to NLP, Language Models

COMP3361 – Week 1

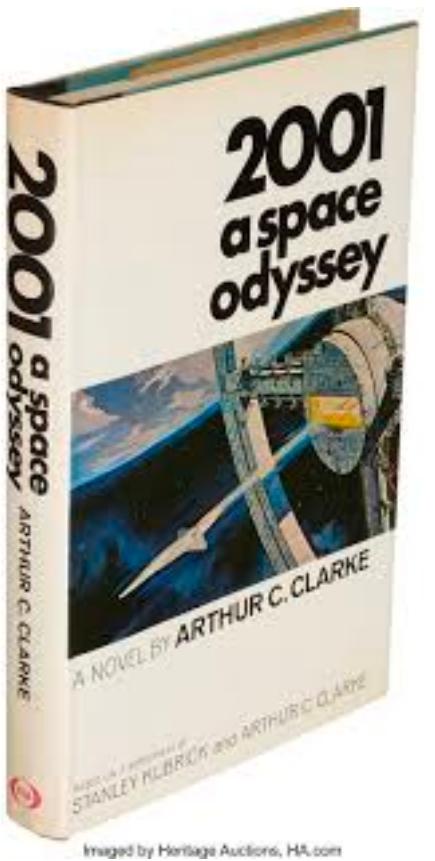
Lingpeng Kong
Department of Computer Science, The University of Hong Kong

Many materials from CSE517@UW, COMS W4705@Columbia, 11-711@CMU with special thanks!

“I'm sorry Dave, I'm afraid I can't do that.”

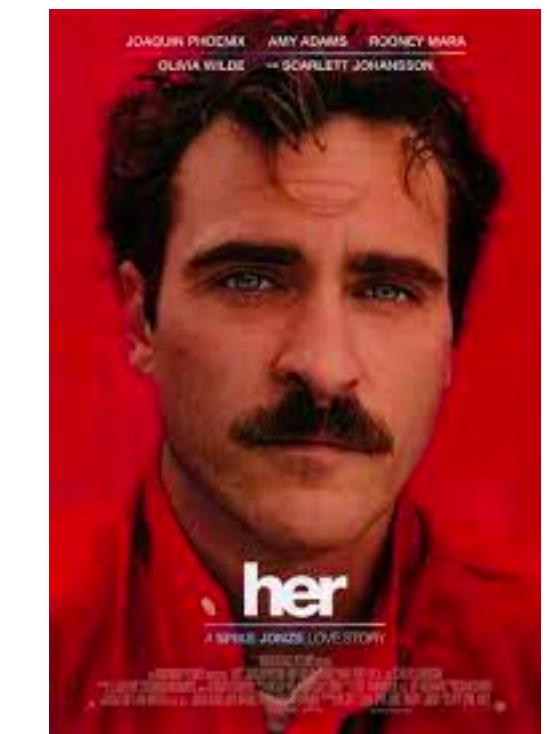


<https://youtube.com/watch?v=ARJ8cAGm6JE>



2001: A Space Odyssey

“I want to learn everything about everything.”



<https://www.youtube.com/watch?v=ne6p6MfLBxc>

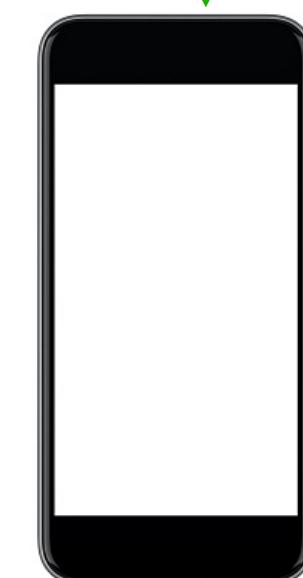
Her (2013)

Talking machines, are we infinitely close or far away?

What can I help you with?

Play a good song.

Sorry, I couldn't find 'a good song' in your music.



You need to do a better job understanding me.

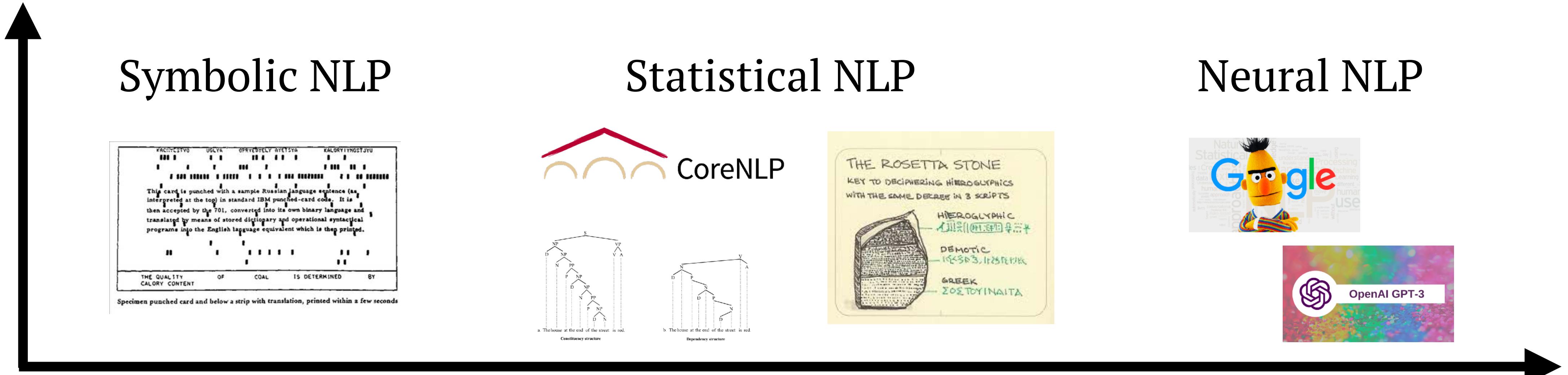
Noted.

Yeah, make a note of that.

Here's your note:



Demystify – what's inside?



1950s - early 1990s

1990s - 2010s

2010s - present

What is NLP? Wait, what is language?

The abstraction of the real world – different languages take you to different worlds!



CX 844
Hong Kong -> New York



餃子

dumplings



CX 596
Hong Kong -> Osaka



Something that makes sharp long voice, like screaming???

嗰吶

チャルメラ

Do talking machines “understand”? Let’s play a game!

The cat is thrown out of the _____

door, window, dog

This year, I am going to do an internship in _____

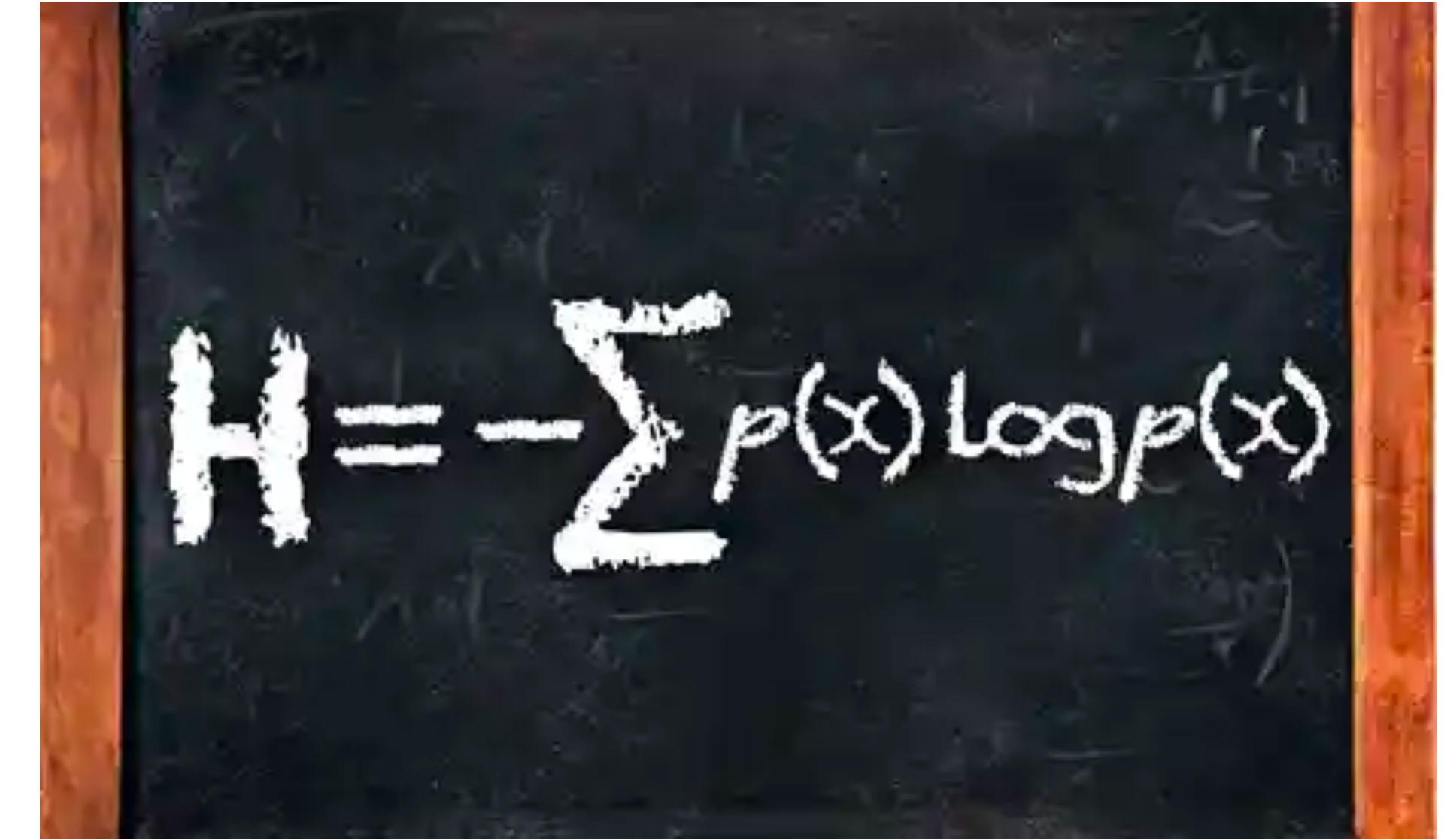
Queen Mary Hospital, HSBC, Google, Amazon

Majoring in computer science, this year, I am going to do an internship in _____

Queen Mary Hospital, HSBC, Google, Amazon

Shannon Game




$$H = -\sum p(x) \log p(x)$$
A photograph of a chalkboard with the mathematical formula for entropy written on it in white chalk. The chalkboard is framed by a wooden border. The formula is $H = -\sum p(x) \log p(x)$.

Information Theory; Entropy

Claude Elwood Shannon
(April 30, 1916 – February 24, 2001)

Language models, and how to build it.



Dice, and how do we roll them
(probabilistic model)



Transformers, neural networks and many others
(powerful functions)

Generative Language Model

I am going to do an internship in Google



Making the dice



bag of words
(@Carnegie Mellon University)

- 1 Belief
- 2 Evidence
- 3 Reason
- 4 Claim
- 5 Think
- 6 Justify
- 7 Also
- ...
- 99 Therefore
- 100 Google

Vocabulary



Generative Language Model

I



I

Generative Language Model

I am



am

Generative Language Model

I am going



going

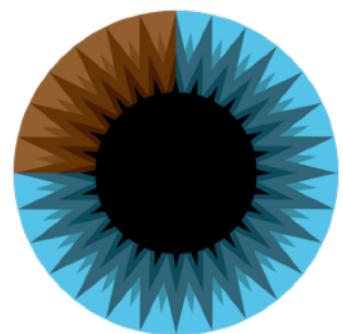
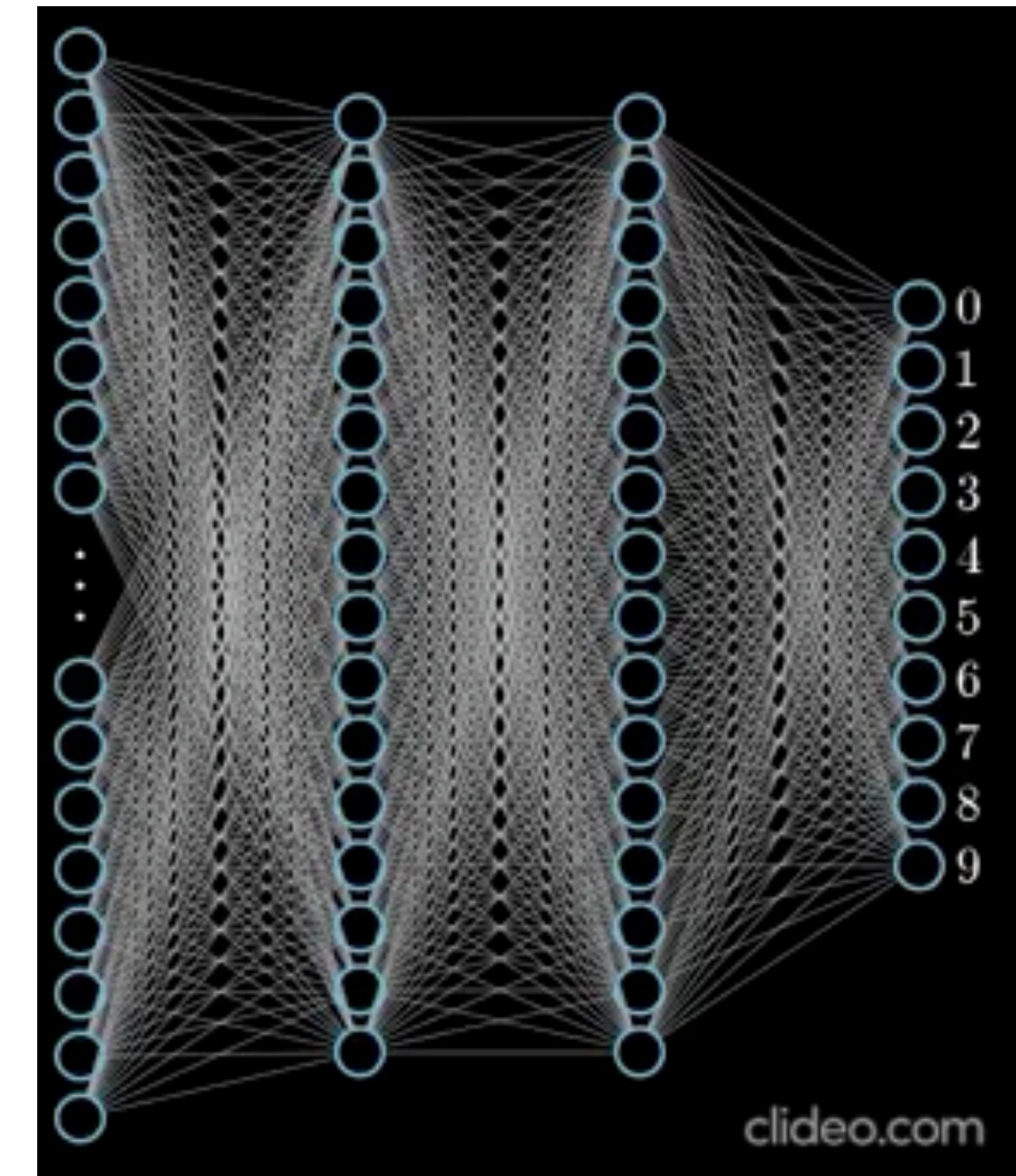
Generative Language Model

I am going to do an internship in Google



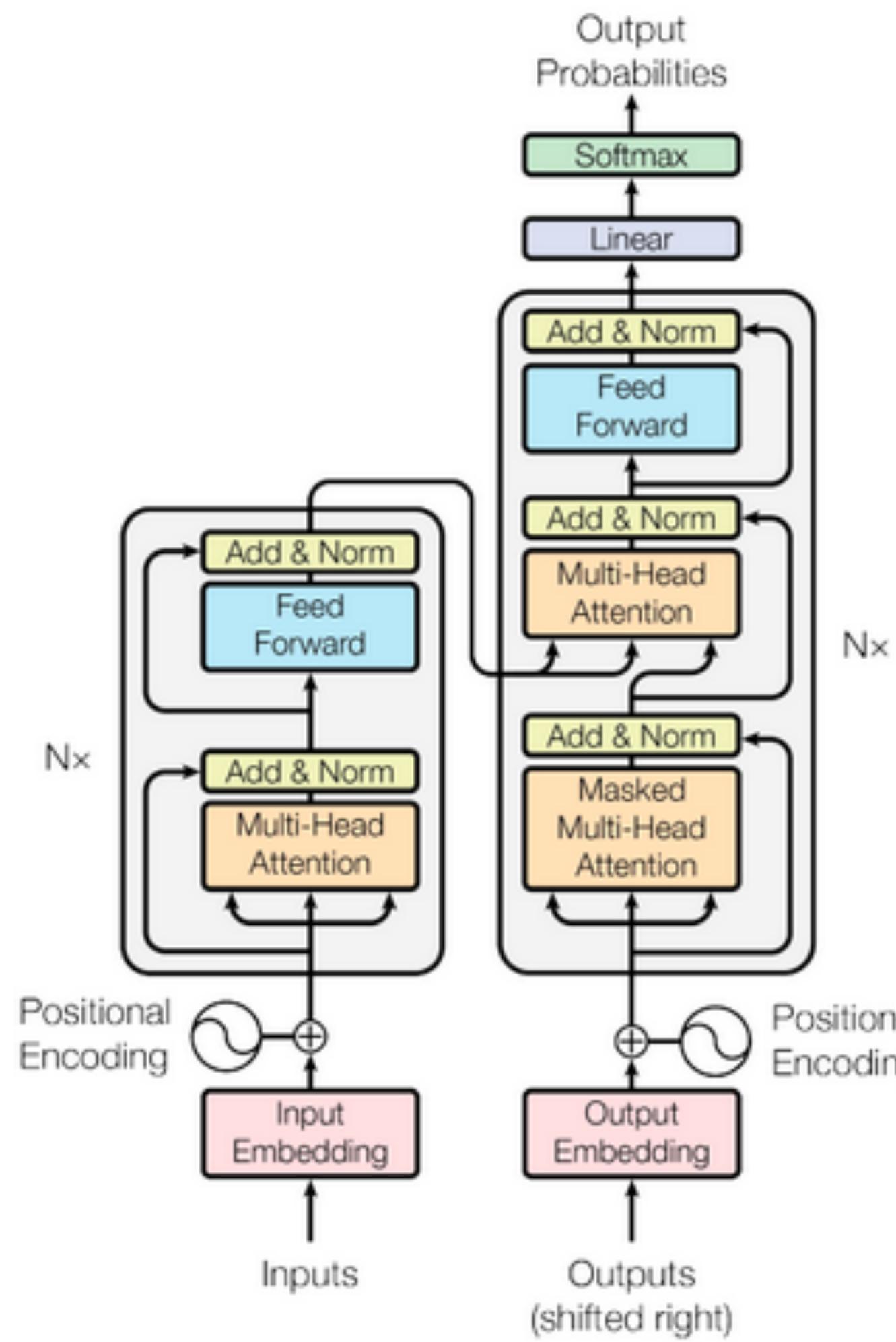
Google

Neuralize the dice!



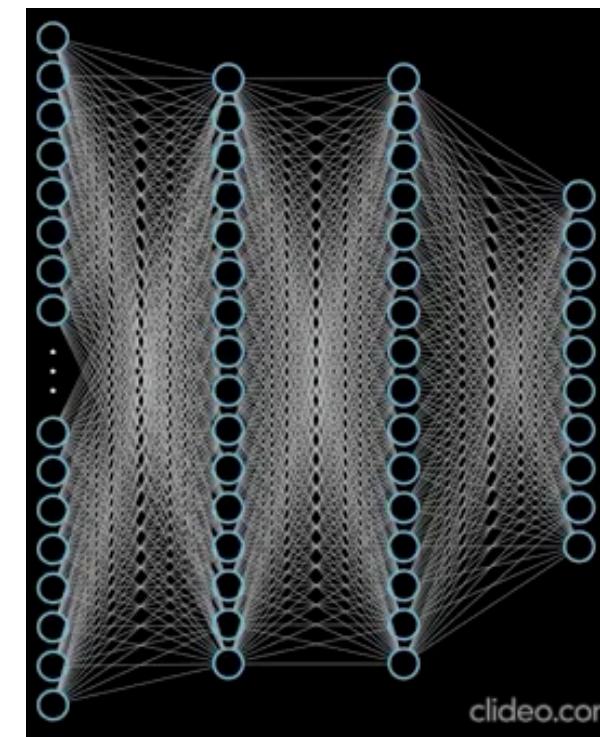
3Blue1Brown

Neural Networks (e.g. Transformers)



Generative Language Model

I am going to do an internship in Google



Google

Language models, and how to build it.

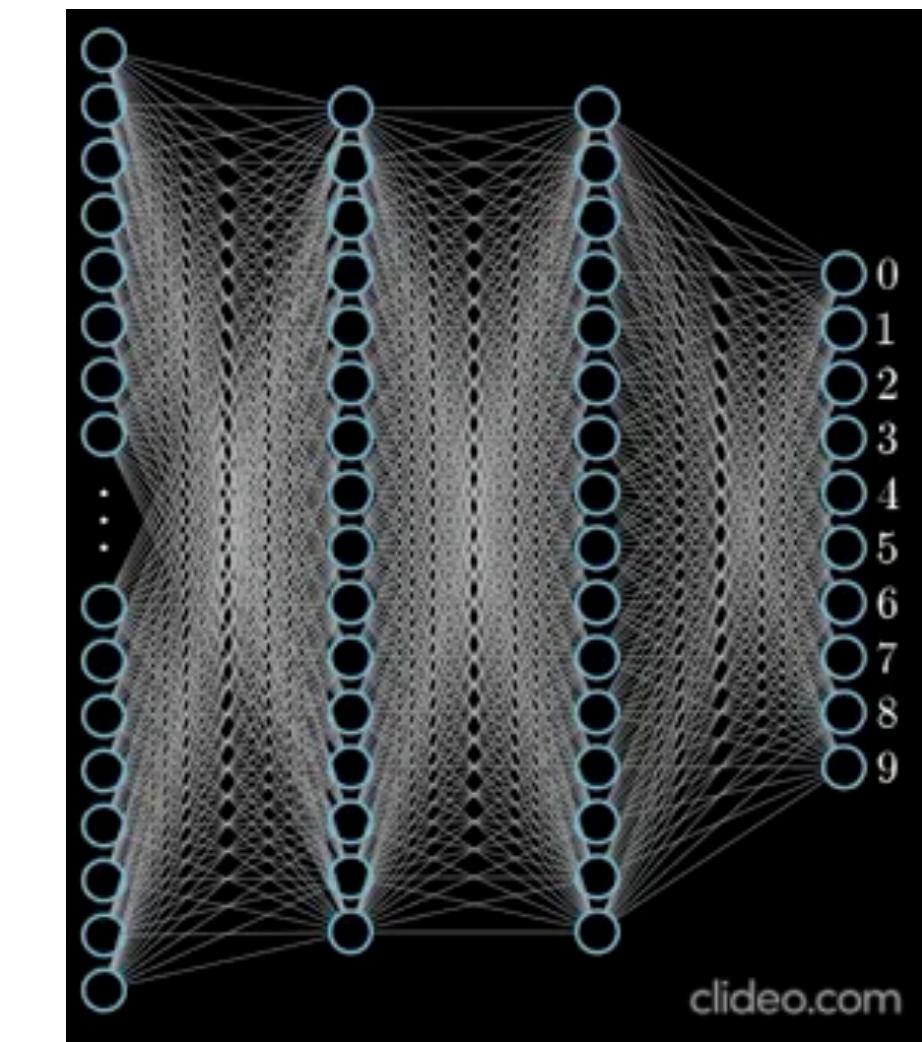
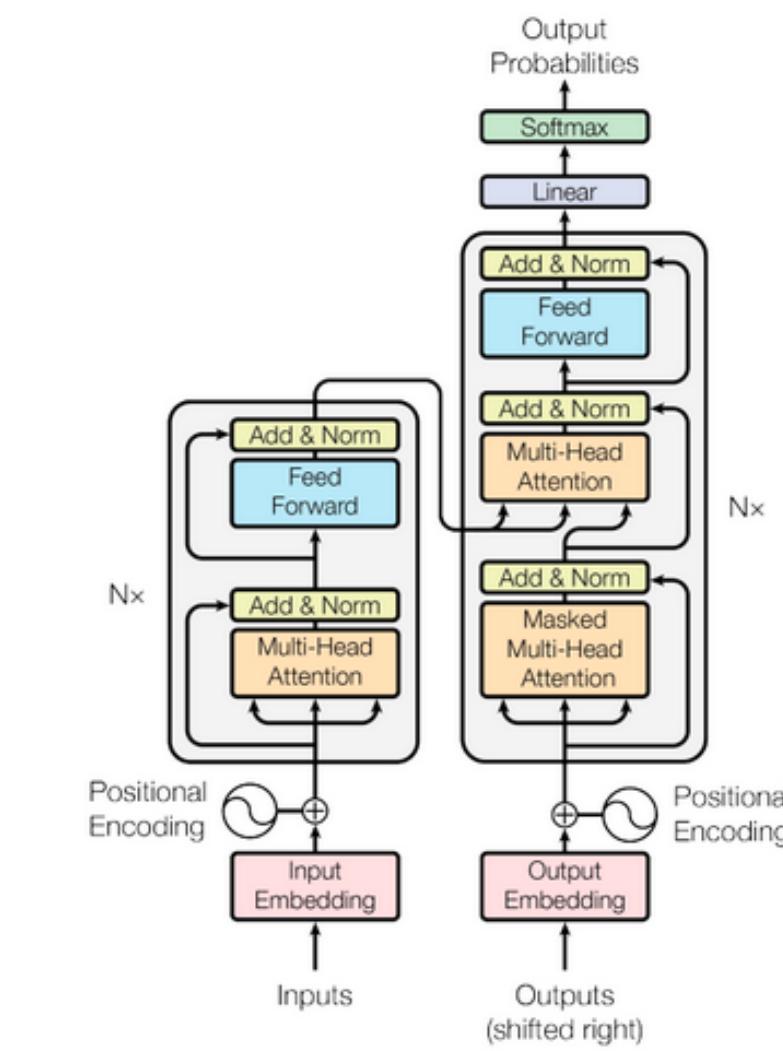
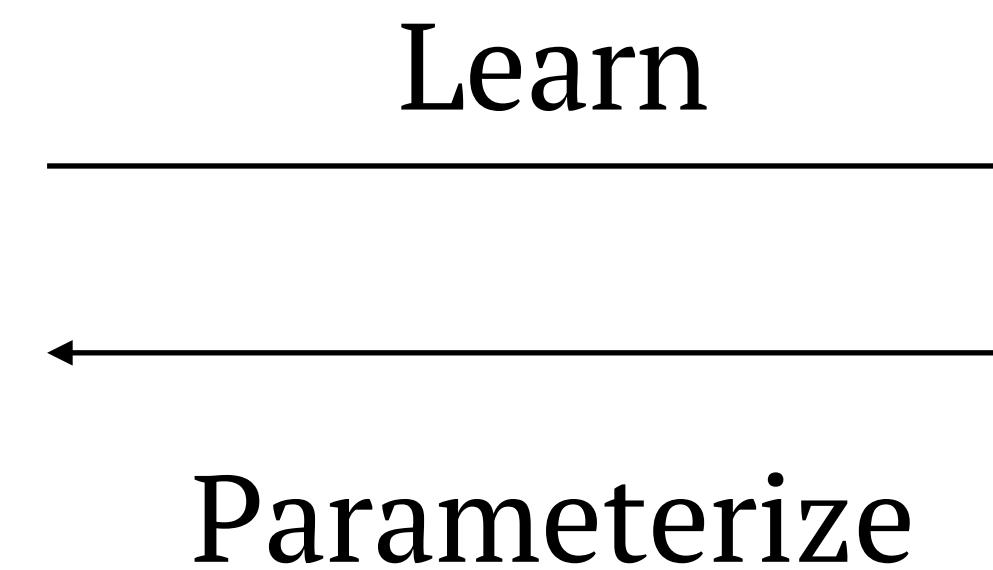


Dice, and how do we roll them
(probabilistic model)



Transformers, neural networks and many others
(powerful functions)

$$p(x) = \prod_i p(x_i | x_{<i})$$



First problem – the language modeling problem

Given a finite vocabulary

$$\mathcal{V} = \{\text{belief, evidence, reason, claim, \dots Google, therefore}\}$$

We have an infinite set of strings, \mathcal{V}^\dagger

< s > I am going to an internship in Google < /s >

< s > an internship in Google < /s >

< s > I am going going < /s >

< s > Google is am < /s >

< s > internship is going < /s >

Can we learn a “model” for this “generative process”? We need to “learn” a probability distribution:

$$\sum_{x \in \mathcal{V}^\dagger} p(x) = 1, \quad p(x) \geq 0 \text{ for all } x \in \mathcal{V}^\dagger$$

Learn from what we've seen

The Language Modeling Problem

Given a *training sample* of example sentences, we need to “learn” a probabilistic model that assigns probabilities to every possible string:

$$p(<\text{s}> \text{ I am going to an internship in Google } </\text{s}>) = 10^{-12}$$

$$p(<\text{s}> \text{ an internship in Google } </\text{s}>) = 10^{-8}$$

$$p(<\text{s}> \text{ I am going going } </\text{s}>) = 10^{-15}$$

...

It is a probability distribution p over strings, i.e., p is a function that satisfies

$$\sum_{x \in \mathcal{V}^\dagger} p(x) = 1, \quad p(x) \geq 0 \text{ for all } x \in \mathcal{V}^\dagger$$

The Language Modeling Problem

<start> Sam I am </start>

<start> I am Sam </start>

<start> I do not like green eggs and ham </start>

Training Corpus

Is this a good model?

$$P(\text{<start> Sam I am </start>}) = 1/3$$

$$P(\text{<start> I am </start>}) = 0$$

$$P(\text{<start> I am Sam </start>}) = 1/3$$

$$P(\text{<start> green Sam </start>}) = 0$$

$$P(\text{<start> I do not like green eggs and ham </start>}) = 1/3$$

Naïve Model

The Language Modeling Problem

< s > Sam I am < /s >

< s > I am Sam < /s >

< s > I do not like green eggs and ham < /s >

Training Corpus

The probability of the word “< s >” followed by the word “I”:

$$P(I | \text{< s >}) = 2/3$$

The Language Modeling Problem

< s > Sam I am < /s >

< s > I am Sam < /s >

< s > I do not like green eggs and ham < /s >

Training Corpus

$$P(I \mid \text{< s >}) = 2/3$$

$$P(\text{< /s >} \mid \text{Sam}) = 1/2$$

Naïve Model

The Language Modeling Problem

< s > Sam I am < /s >

< s > I am Sam < /s >

< s > I do not like green eggs and ham < /s >

Training Corpus

$$P(I \mid \text{< s >}) = 2/3$$

$$P(\text{am} \mid I) = ?$$

$$P(\text{< /s >} \mid \text{Sam}) = 1/2$$

Naïve Model

The Language Modeling Problem

< s > Sam I am < /s >

< s > I am Sam < /s >

< s > I do not like green eggs and ham < /s >

Training Corpus

$$P(I \mid \text{< s >}) = 2/3$$

$$P(\text{am} \mid I) = 2/3$$

$$P(\text{< /s >} \mid \text{Sam}) = 1/2$$

Naïve Model

The Language Modeling Problem

< s > Sam I am < /s >

< s > I am Sam < /s >

< s > I do not like green eggs and ham < /s >

Training Corpus

$$P(< s > \text{ Sam I am } < /s >) = P(\text{Sam} | < s >) * P(\text{I} | \text{Sam}) * P(\text{am} | \text{I}) * P(< s > | \text{am})$$

Bi-gram Model

Course Logistics

Course Logistics

Website:

<https://nlp.cs.hku.hk/comp3361>

Prerequisites:

COMP3314 or COMP3340, MATH1853

We will assume a lot things from Machine Learning, Statistics, and Programming



This NLP course will be very difficult if you haven't taken these courses.

Assessment:

50% continuous assessment, 50% examination

TA:

Qintong Li (<https://qtli.github.io/>)

Course Logistics

Textbook recommendation (J&M):

<https://web.stanford.edu/~jurafsky/slp3/>

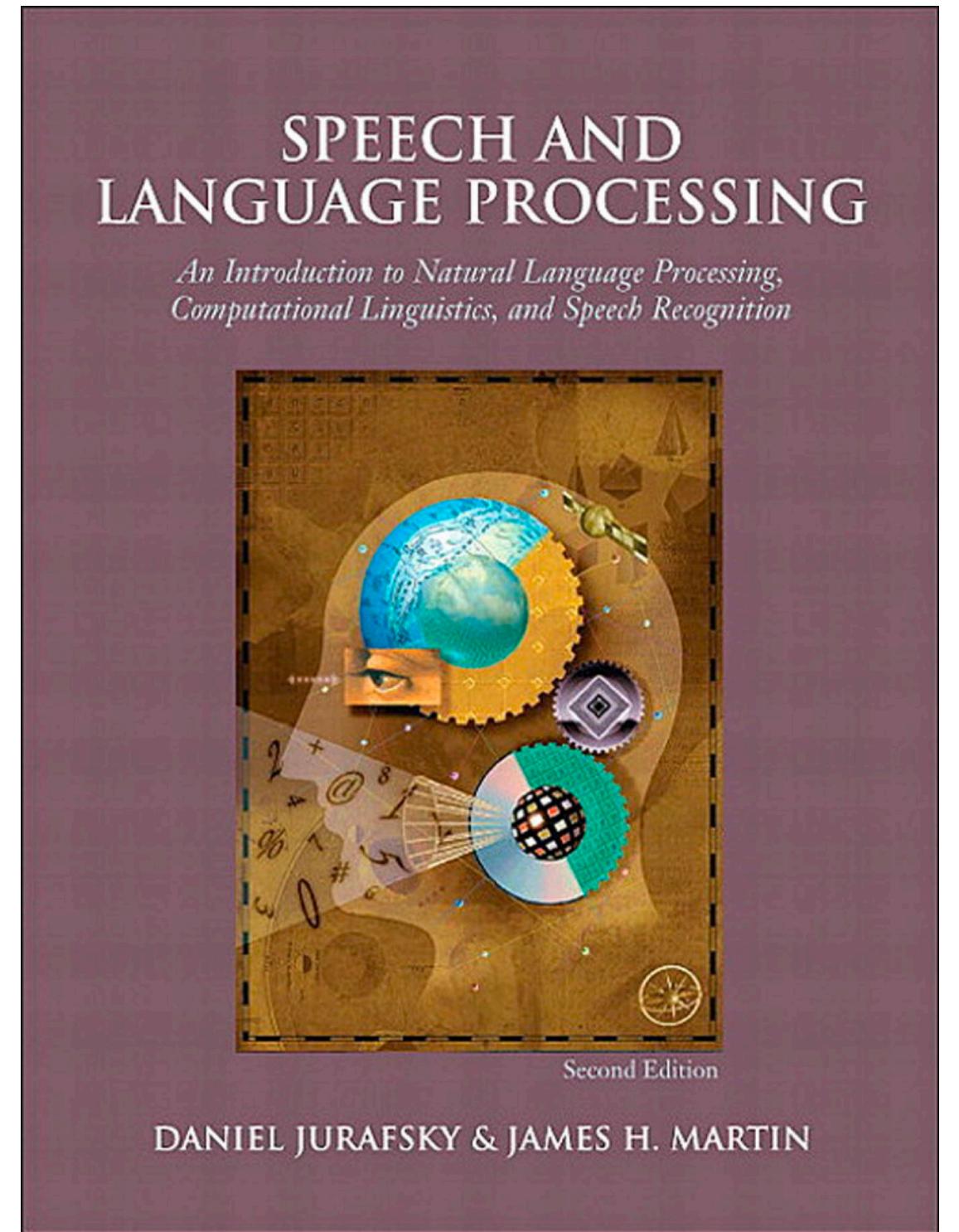
Assessments (in total 4, each 25%):

Programming problems

Problem sets

Honor code:

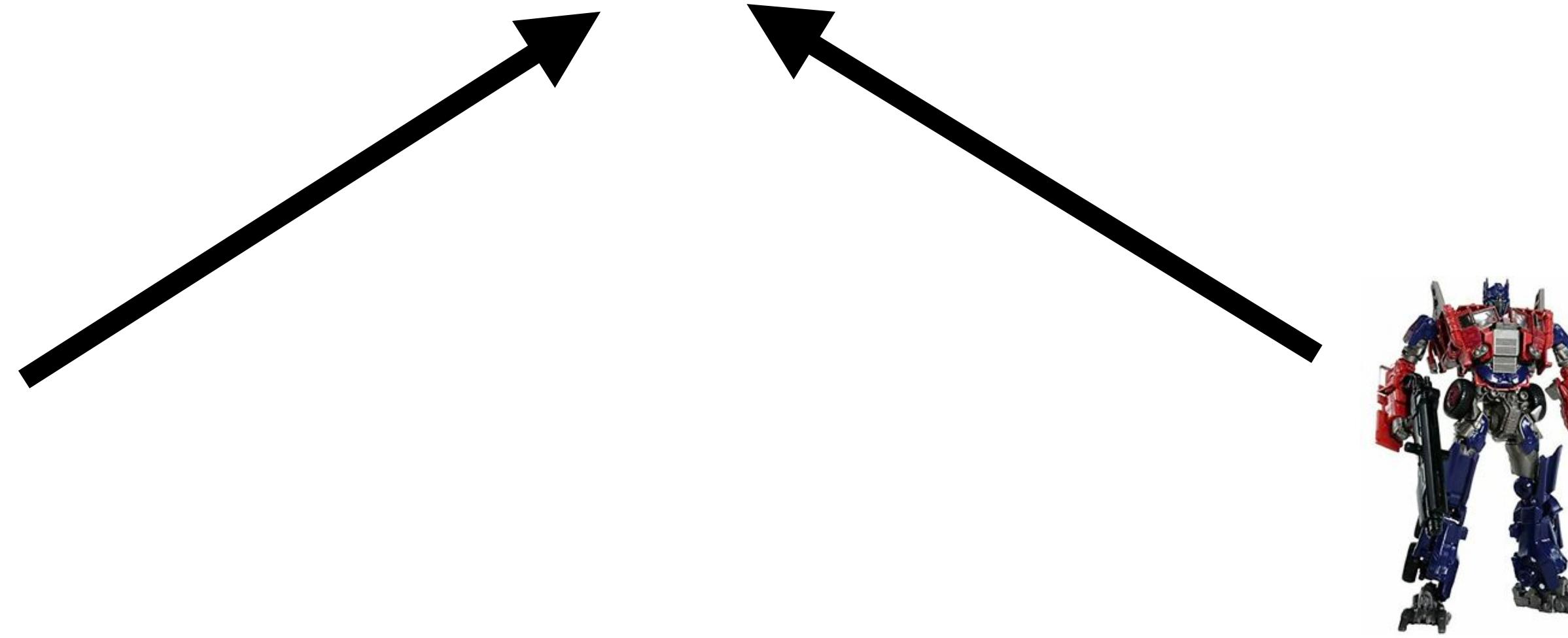
You are free to form study groups and discuss homeworks and projects. However, you must write up homeworks and code from scratch independently, and you must acknowledge in your submission all the students you discussed with.



What's Next?



BERT, GPT-3, Word2vec, Glove, T5 ...

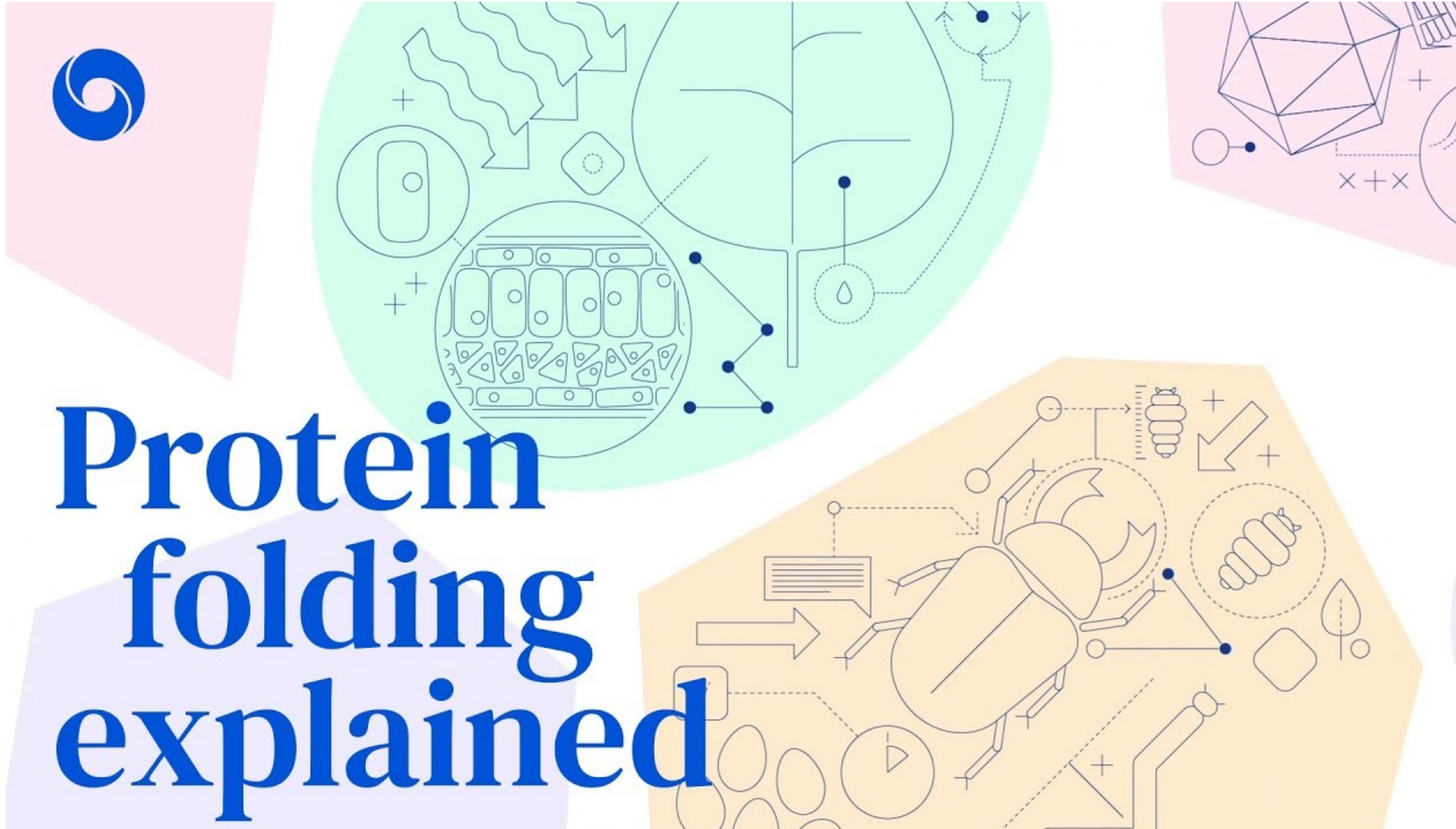


N-Gram Models, Hidden Markov Models ...

LSTMs, Recurrent Neural Networks, MLP, Transformers ...

It's not about human language. It's the language of life!

Protein folding explained



 Google DeepMind

<https://www.youtube.com/watch?v=KpedmJdrTpY>

AlphaFold (Jumper et al, 2021)