Wrapping up discussion of “312 meets LLMs”
A glimpse of auction theory

No more concept checks!
Review in class on Friday

CSE 312 Spring 26
Lecture 26

The central object in a language model

$$\boxed{\mathbb{P}(\text{next token} \mid \text{context})}$$

- **Token:** A **word** or part of a word or punctuation.
- **Context:** prompt, previous words output and more

An entire response is generated one conditional probability at a time.

- You enter prompt
- LLM computes $\mathbb{P}(w_1 \mid \text{prompt, any other context})$.
- Samples from this distribution
- LLM computes $\mathbb{P}(\, w_2 \mid w_1 \text{ , prompt, any other context}\,)$
- Samples from this distribution.
- And so on.

Distribution over outputs (by chain rule)

$$\mathbb{P}(w_1, w_2, \dots, w_n \mid \text{prompt}, \text{context})$$
$$= \mathbb{P}(w_1 \mid \text{prompt}) \cdot \mathbb{P}(w_2 \mid w_1, \text{prompt}) \cdot \mathbb{P}(w_3 \mid w_1, w_2) \dots \mathbb{P}(w_n \mid w_1, \dots, w_{n-1})$$

This simple idea, scaled up with

- **huge data sets**
- **huge neural networks**
- **massive amounts of computation**

leads to ChatGPT, Gemini, etc.…

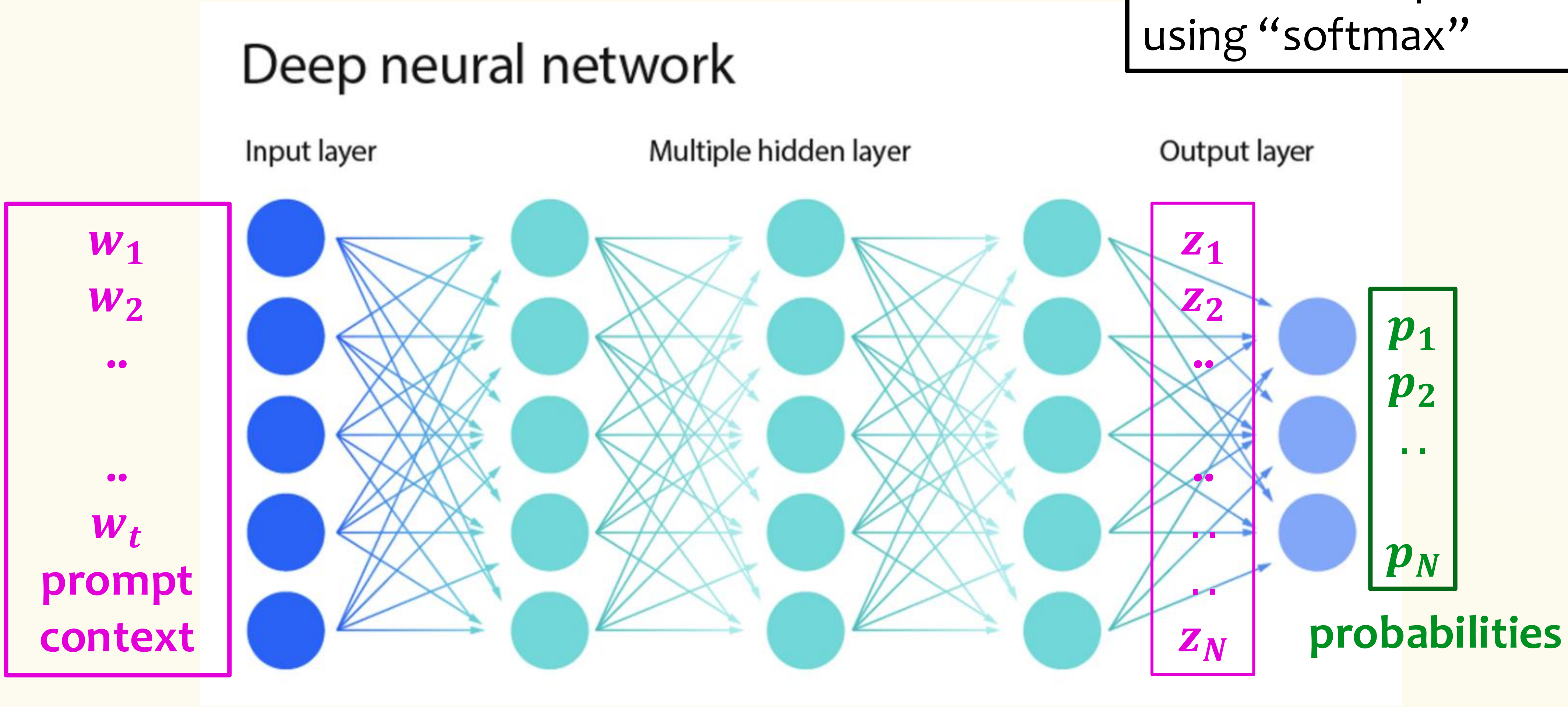
Some details to work out:

Where do these probabilities come from? What is the sampling process?
And much more….

Where do the probabilities come from?

How does the LLM compute $\mathbb{P}(w_t \mid w_1, \ldots, w_{t-1}, \mathrm{prompt}$

Logits are real numbers
Converted to probabilities
using “softmax”



inputs

logits (scores)

Logits -> probabilities via softmax

$\mathbb{P}$(next word | "The probabilities of all disjoint outcomes must sum to")

- $z_{\text{one}} = 2.0, \quad z_{\text{zero}} = 1.0, \quad z_{\text{infinity}} = 0.0, \quad z_{\text{pizza}} = -1.0$

$$\boxed{p_i = \frac{e^{z_i}}{\sum_{j=1}^{V} e^{z_j}}}$$

- $p_{\text{one}} = 0.64, \quad p_{\text{zero}} = 0.24, \quad p_{\text{infinity}} = 0.09, \quad p_{\text{pizza}} = 0.03.$

$$\mathbb{P}(w_1, w_2, \dots, w_n \mid \text{prompt, context})$$
$$= \mathbb{P}(w_1 \mid \text{prompt}) \cdot \mathbb{P}(w_2 \mid w_1, \text{prompt}) \cdot \mathbb{P}(w_3 \mid w_1, w_2) \dots \mathbb{P}(w_n \mid w_1, \dots, w_{n-1})$$

Logits -> Probabilities -> Sampling

- To allow for "tuning" of this probability distribution, a temperature parameter *T* is introduced

$$\boxed{p_i = \frac{e^{z_i/T}}{\sum_{j=1}^{V} e^{z_j/T}}}$$

Logits -> probabilities via softmax

$\mathbb{P}$(next word | "The probabilities of all disjoint outcomes must sum to")

$z_{\text{one}} = 2.0, \quad z_{\text{zero}}=1.0, \quad z_{\text{infinity}}=0.0, \quad z_{\text{pizza}} = -1.0$

$$\boxed{p_i = \frac{e^{z_i/T}}{\sum_{j=1}^{V} e^{z_j/T}}}$$

T = 1. $p_{\text{one}} = 0.64, \quad p_{\text{zero}} = 0.24, \quad p_{\text{infinity}} = 0.09, \quad p_{\text{pizza}} = 0.03.$

T = 0.5. $p_{\text{one}} = 0.87, \quad p_{\text{zero}} = 0.12, \quad p_{\text{infinity}} = 0.015, \quad p_{\text{pizza}} = 0.001.$

Logits -> Probabilities -> Sampling

- To allow for "tuning" of this probability distribution, a temperature parameter *T* is introduced

$$\boxed{p_i = \frac{e^{z_i/T}}{\sum_{j=1}^{V} e^{z_j/T}}}$$

- What happens when increase the temperature?
  – distribution becomes uniform!
- What happens when decrease the temperature?
  – Pushes probability towards more likely words
  – As T tends to 0, goes towards "greedy decoding".

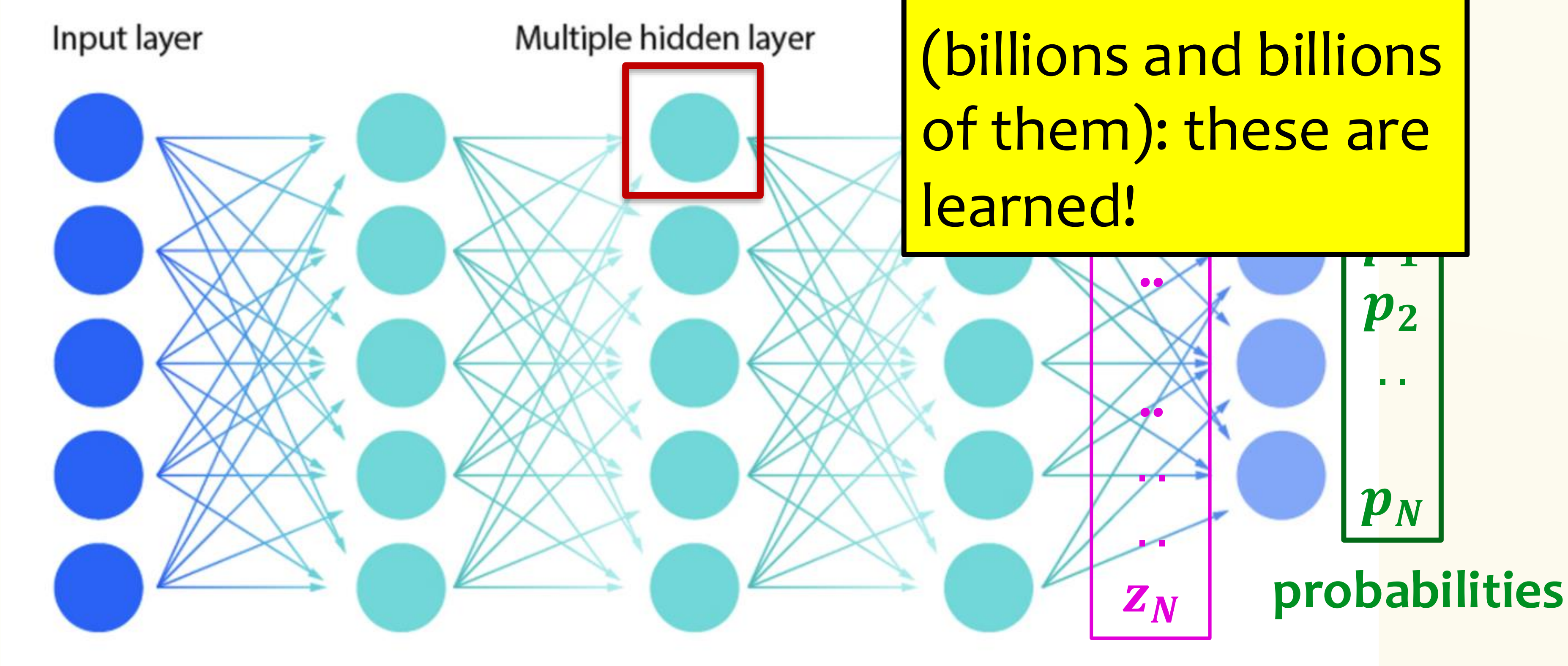
Back to the neural network: what's going on inside

Typical node: takes inputs $x_1, \dots, x_k$ and outputs $f(\, w_1 x_2 + \cdots + w_k x_k + b)$ where f is a non − linear "activation" function.

w's and b's called weights and biases

The parameters of the neural network (billions and billions of them): these are learned!

Training dataset: massive corpus



Pre-training

- For months and months, collect all the text they can: Wikipedia, books, articles, code, Internet forums
- Goal: is to set the parameters θ (i.e., weights and biases) of the network to maximize the likelihood of seeing the text in the training set

$$\mathcal{L}(\text{corpus};\ \theta) = \prod_{\text{i}=1}^{\text{N}} \mathbb{P}_\theta(\text{ token } i \mid \text{previous tokens })$$

Actually, minimize negative log likelihood

$$-\ln \mathcal{L}(\text{corpus};\ \theta) = -\sum_{i=1}^{N} \mathbb{P}_\theta(\text{ token } i \mid \text{previous tokens })$$

The Pre-training Loop

Repeat billions of times:

- Read a chunk of text from the Internet
- Use the current weights to guess the probability of the next word
- Check what the actual next word was
- Calculate the negative log likelihood of seeing that word.
- Use calculus (“Backpropogation”) to figure out which direction to nudge every single parameter (weight and bias) to make that specific word slightly more probable next time.

When done: you have a Base Model! (pre-trained)

BUT… not useful yet!

Making the model useful/helpful

- Maximize human preference rather than pure likelihood.

- How to learn human preference?
- Ask humans!

The helpfulness problem

Suppose one of you asks our 312 GPT for help with a problem.

The assistant generates two possible hints:

- Hint A: Start by identifying the sample space. Then define the event you are counting.
- Hint B: The answer is $\binom{52}{5}$

Different humans will differ on which hint they prefer.

Helpfulness is not objective.

The Bradley-Terry Model

Basic assumption: for every possible response, there is a hidden "helpfulness score", we call it R (for reward) – could be positive or negative.

Unknown to us.

Can learn it based on preferences expressed by human.

If we are comparing two responses A and B,

R_A = hidden score of response A, . R_B = hidden score of response B

Assume that $\mathbb{P}(A \succ B) = \frac{e^{R_A}}{e^{R_A}+e^{R_B}}$.

Use MLE to estimate the parameters from the preferences expressed by humans!

Reinforcement Learning from Human Feedback (RLHF) workflow

- Companies hire armies of human contractors.
  - Sit at computers, read a prompt, read two AI-generated responses and click "A is better" or "B is better", thousands of times a day.
- Train a Reward Model.
  - If 10 people rate a particular pair of responses, and 8 people prefer A, say $\mathcal{L}(R_A, R_\mathrm{B}) = \left(\frac{e^{R_A}}{e^{R_A+e^{R_B}}}\right)^8 \left(\frac{e^{R_\mathrm{B}}}{e^{R_A+e^{R_B}}}\right)^2$
  - Find the parameters R_A, R_B that make our observed data most likely
  - Once trained, it can estimate how helpful a human would find any new text.

Reinforcement Learning step

Bring Reward Model back to Language Model:

“Your new goal in life is to get the highest score possible from the Reward Model.”

- AI generates thousands of responses.
- Reward Model grades them.
  - When it gets a high score, it updates internal parameters to behave that way more often.
  - When it gets a low score, adjusts parameters to avoid that behavior.

Some takeaways

- A language model repeatedly estimates a conditional distribution of the form: $\boxed{\mathbb{P}(\text{next token} \mid \text{previous tokens})}$
- The chain rule for probability is the mathematical reason we can assign probabilities to whole sequences one token at a time.
- Maximum likelihood training rewards a model for assigning high probability to the actual observed next token and to text that humans find helpful.
- Sampling from a distribution explains why LLM outputs can be useful, creative, variable and sometimes wrong (aka hallucinations).

Agenda

- CSE 312 meets LLMs
- A glimpse of auction theory

Auctions

- Some goods on eBay and amazon are sold via auction.
- Companies like Google and Meta (and many others) make most of their money by selling ads.

Facebook Ads bidding... 🤔 Is this an auction?

Yes! That's the first thing you need to understand to master bidding management of Facebook Ads. **When you're creating a new campaign, you're joining a huge, worldwide auction.**

You'll be competing with hundreds of thousands of advertisers to buy what Facebook is selling: Real estate on the News Feed, Messenger, Audience Network, and mobile apps to display your ads to the users.

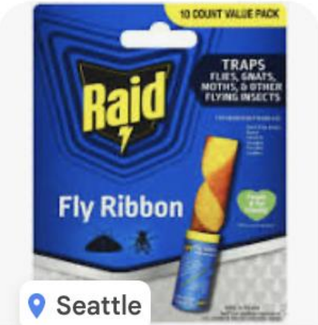
how to get rid of fruit flies

ools

- **Empty Trash and Recycling:** Regularly take out the kitchen trash and rinse out recycling bins—especially beer, wine, and juice containers. Reddit · r/pestcontrol +1
- **Clean the Drains:** Fruit flies love to breed in the slime and food residue inside sink drains. Pour boiling water down your drains, or use a mixture of white vinegar to kill hidden eggs. Reddit · r/pestcontrol +1

Sponsored

Here are some related products to consider:



Seattle

Raid Fly & Bug Catcher Ribbons...

$2.99

QFC

TERRO Fruit Fly Traps - 0.675 fl oz

$5.99

Instacart

Fruit Fly Traps for Indoors Mosquito...

$19.98 ~~$38~~

LightInTheBox

Plug-In Flying Insect Trap for...

$19.99

Pest Guard

Auctions

- The ads are sold via auction.
    - Advertisers submit bids for certain “keywords”.

Facebook Ads bidding… 🤔 Is this an auction?

Yes! That’s the first thing you need to understand to master bidding management of Facebook Ads. **When you’re creating a new campaign, you’re joining a huge, worldwide auction.**

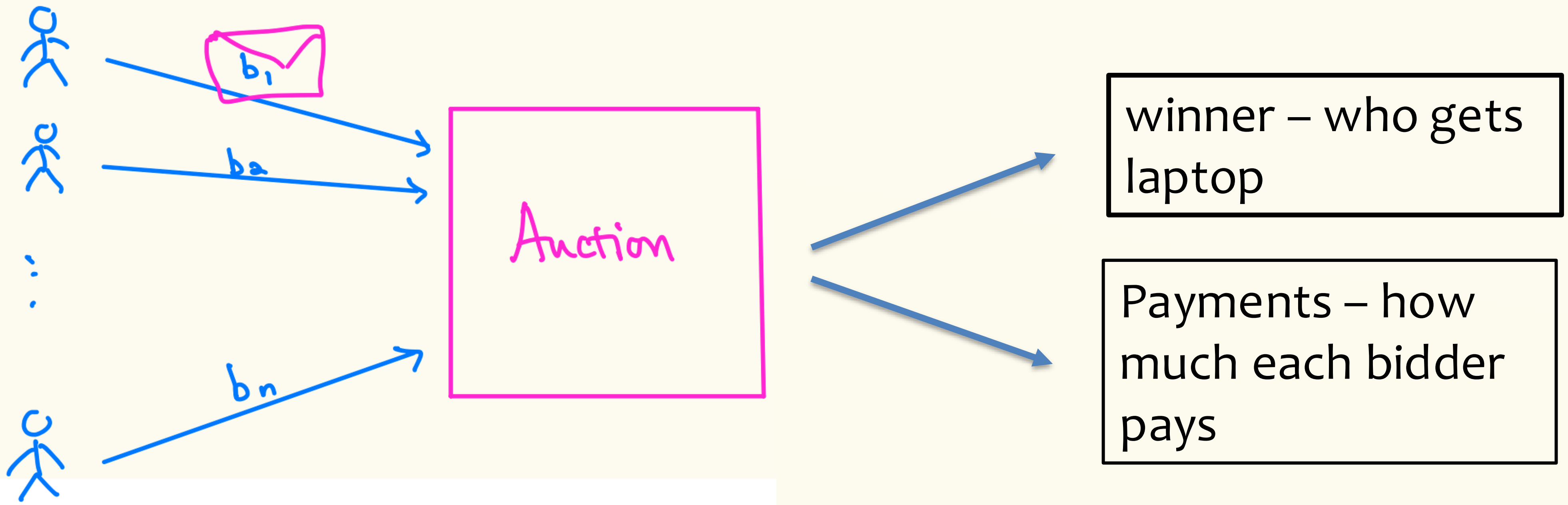
You’ll be competing with hundreds of thousands of advertisers to buy what Facebook is selling: Real estate on the News Feed, Messenger, Audience Network, and mobile apps to display your ads to the users.

An auction is a …

- Game
  - Players: advertisers
  - Strategy choices for each player: possible bids
  - Rules of the game – made up by Google/Meta/ whoever is running the auction
- What do we expect to happen? How do we analyze mathematically?

Special case: Sealed bid single item auction

- Say I decide to run an auction to sell my laptop and I let you be the bidders.
- If I want to make as much money as possible – what should I choose as the rules of the auction?



Auction defines a mapping from bids to a winner and payments

Special case: Sealed bid single item auction

- Say I decide to run an auction to sell my laptop and I let you be the bidders.
- If I want to make as much money as possible – what should the rules of the auction be?

Some possible auction formats:

- **First price auction:** highest bidder wins; pays what they bid.
- **Second price auction:** highest bidder wins; pays second highest bid.
- **All pay auction:** highest bidder wins: all bidders pay what they bid.

Which of these will make me the most money?

Sealed Bid single item auctions:

- **First price auction:** highest bidder wins; pays what they bid.
- **Second price auction:** highest bidder wins; pays second highest bid.
- **All pay auction:** highest bidder wins: all bidders pay what they bid.

Bidder	1	2	3	4
Bids	**100**	**81**	**35**	**24**
First price	100	0	0	0
Second price	81	0	0	0
All pay	100	81	35	24

Bidder model

Each bidder has a value, say v_i for bidder i.

Bidder is trying to maximize their "utility" –
the value of the item they get – price they pay.

value -		**price**	**=**	**utility**
$100	-	$86	=	$14
$100	-	$102	=	$-2

Let's compare

- First price auction
- Second price auction
- All pay auction

When there are 2 bidders and their values are drawn uniformly from [0, 100]

How will they bid and what will make me the most money?

Theorem

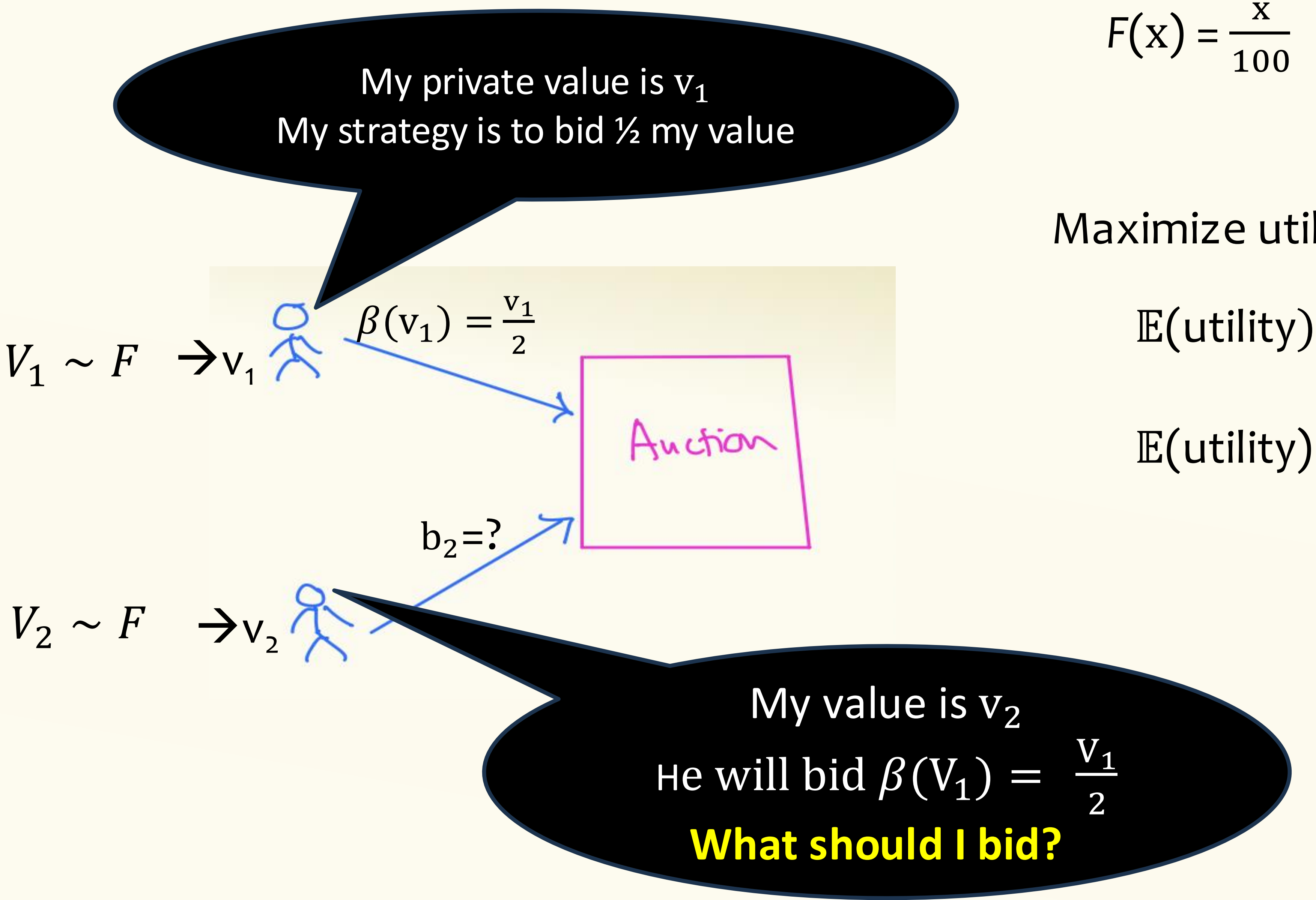
A second price auction is **truthful**. In other words, it is always in each bidder's best interest to bid their true value.

First price auction – 2 bidders

$V_1, V_2 \sim \mathrm{F} = Unif[0,100]$

Independent

$F(\mathrm{x}) = \frac{\mathrm{x}}{100}$



Maximize utility!

$$\mathbb{E}(\text{utility}) = (v_2 - b_2)\,\mathbb{P}(\text{I win})$$

$$\mathbb{E}(\text{utility}) = (v_2 - b_2)\,\mathbb{P}\left(b_2 \geq \frac{V_1}{2}\right)$$

$$= (v_2 - b_2)\,\mathbb{P}\,(V_1 \leq 2\,b_2)$$

$$= (v_2 - b_2)\frac{2b_2}{100}$$

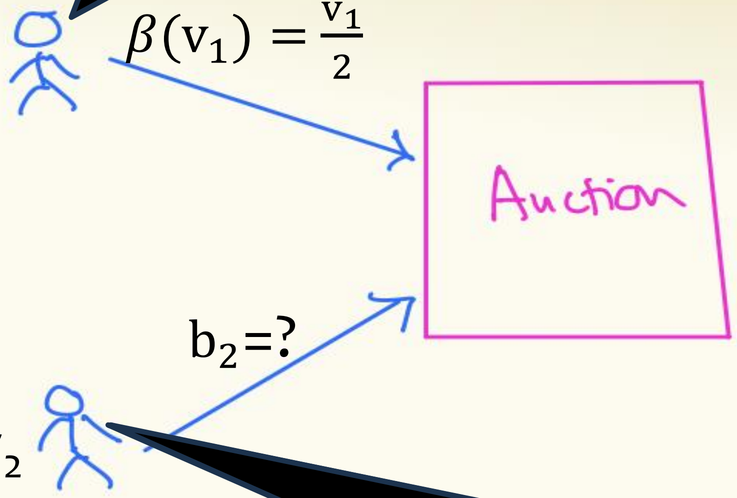
Bayes-Nash equilibrium
First price auction – 2 bidders

$V_1, V_2 \sim Unif[0,100]$
Independent
F(x) = x/100.

My private value is v_1
My strategy is to bid ½ my value

$\beta(v_1) = \frac{v_1}{2}$

$V_1 \sim F \rightarrow v_1$



b_2=?

$V_2 \sim F \rightarrow v_2$

My value is v_2
He will bid $\beta(V_1) = \frac{V_1}{2}$
What should I bid?

$\mathbb{E}$(utility if bid b_2) $= (v_2 - b_2)\frac{2b_2}{100}$

expectation w.r.t random variable V_2

Choose b_2 to maximize $(v_2 - b_2)\frac{2b_2}{100}$

Differentiate w.r.t. b_2 and set = 0

Solution: $b_2 = \frac{v_2}{2}$.

When the values are uniform [0,100]

- 2 bidders participating in a first-price auction.
- If each of them bids half of their value, they are “best-responding” to what the other one is doing (in expectation)
- This is called an equilibrium, or, more precisely, a Bayes-Nash equilibrium.

2 bidders with independent uniform values

$V_1, V_2 \sim \mathrm{Unif}[0,100]$

$\beta(\cdot)$ is "equilibrium" bidding strategy

	2^{nd} price	1^{st} price	All pay auction
V	$\boldsymbol{\beta(V) = V}$	$\boldsymbol{\beta(V) = \frac{V}{2}}$	$\boldsymbol{\beta(V) = \frac{V^2}{200}}$
Exp Auctioneer Revenue	$\mathbb{E}(\min(V_1, V_2))$	$\mathbb{E}(\max(V_1/2, V_2/2))$	$\mathbb{E}(V_1^2/200 + V_2^2/200))$
	$\boxed{=100/3}$	$=(1/2)\,\mathbb{E}(\max(V_1, V_2))$ $=(1/2)\, 2\cdot 100/3$ $\boxed{=100/3}$	$\boxed{=100/3}$

Revenue Equivalence Theorem

In equilibrium, no matter what distribution the bids are drawn from, the expected auctioneer revenue is the same in all three auctions!

This is part of Nobel prize winning work in economics!!