TASK 2

Group members: Sonal Mogra & Zhihuang Chen

The prompt

- The "prompt" usually refers to a piece of text or input that is given to the model to produce some kind of output or response. Essentially, a "hint" is a way to communicate with or guide the model to perform a required task.

Name	Notation	Example	Description
Input	$\boldsymbol{x}$	I love this movie.	One or multiple texts
Output	$\boldsymbol{y}$	++ (very positive)	Output label or text
Prompting Function	$f_{\text{prompt}}(\boldsymbol{x})$	`[X]` Overall, it was a `[Z]` movie.	A function that converts the input into a specific form by inserting the input $\boldsymbol{x}$ and adding a slot `[Z]` where answer $\boldsymbol{z}$ may be filled later.
Prompt	$\boldsymbol{x}'$	I love this movie. Overall, it was a `[Z]` movie.	A text where `[X]` is instantiated by input $\boldsymbol{x}$ but answer slot `[Z]` is not.
Filled Prompt	$f_{\text{fill}}(\boldsymbol{x}', \boldsymbol{z})$	I love this movie. Overall, it was a bad movie.	A prompt where slot `[Z]` is filled with any answer.
Answered Prompt	$f_{\text{fill}}(\boldsymbol{x}', \boldsymbol{z}^*)$	I love this movie. Overall, it was a good movie.	A prompt where slot `[Z]` is filled with a true answer.
Answer	$\boldsymbol{z}$	"good", "fantastic", "boring"	A token, phrase, or sentence that fills `[Z]`

Prompt structure

- zero-shot: Prompt

```
Translate English to French:
cheese =>
```

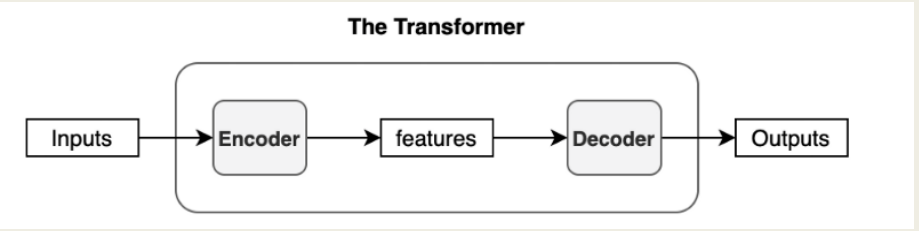
- one-shot: Prompt Augmentation

```
Translate English to French:
sea otter => loutre de mer
cheese =>
```

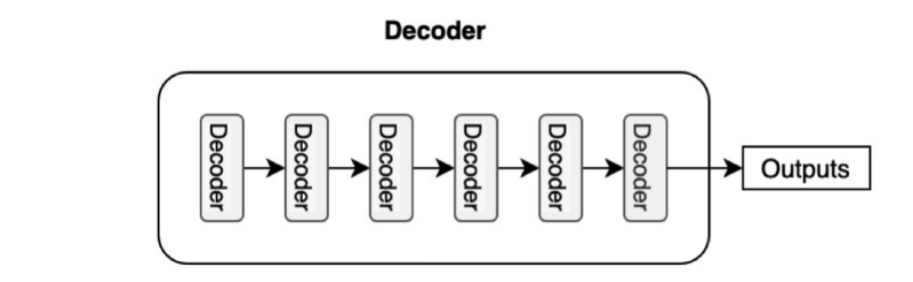
- few-shot: Prompt Augmentation

```
Translate English to French:
sea otter => loutre de mer
peppermint => menthe poivrée
plush girafe => girafe peluche
cheese =>
```

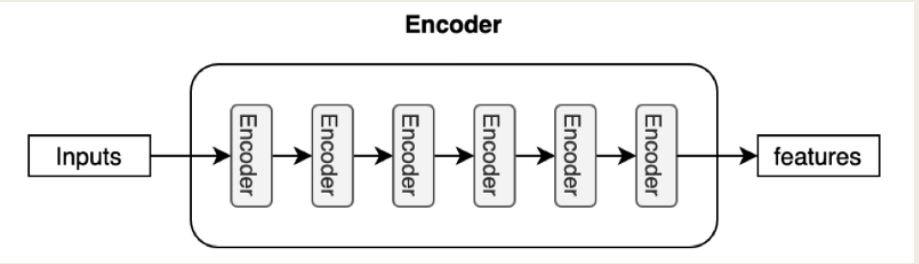
Transformer



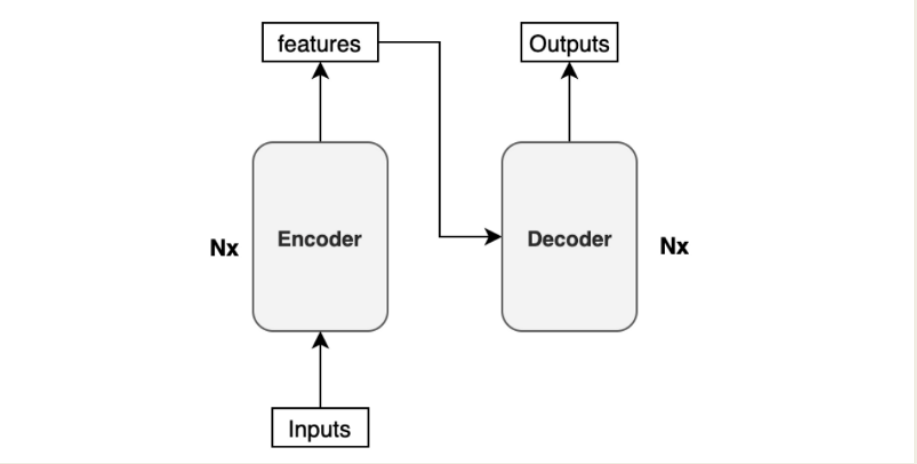
Transformer
- Decoder



Transformer
- Encoder



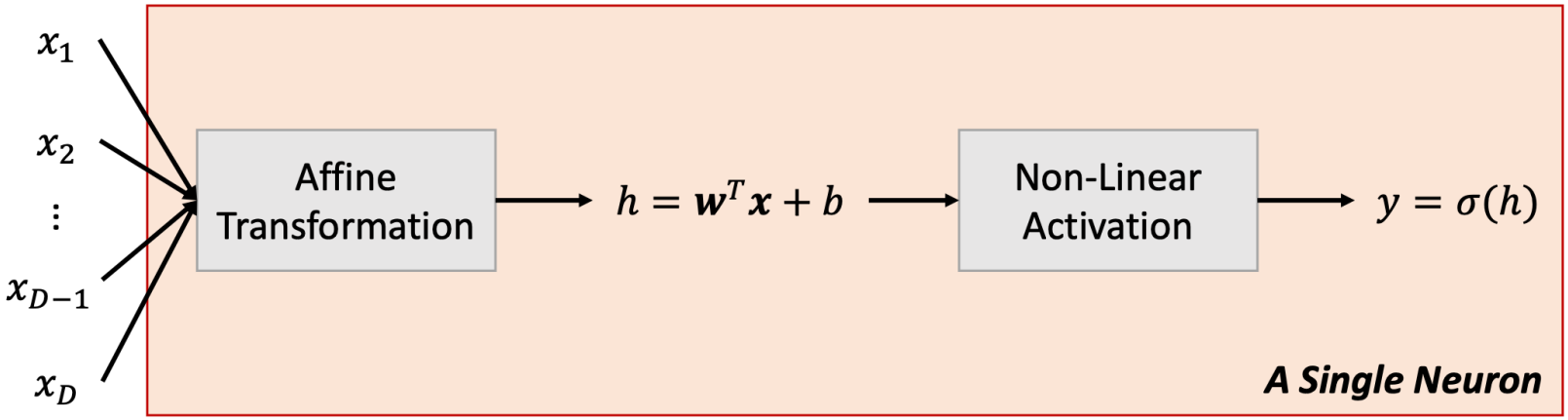
Transformer
-Encoder & Decoder



Single Neuron Network

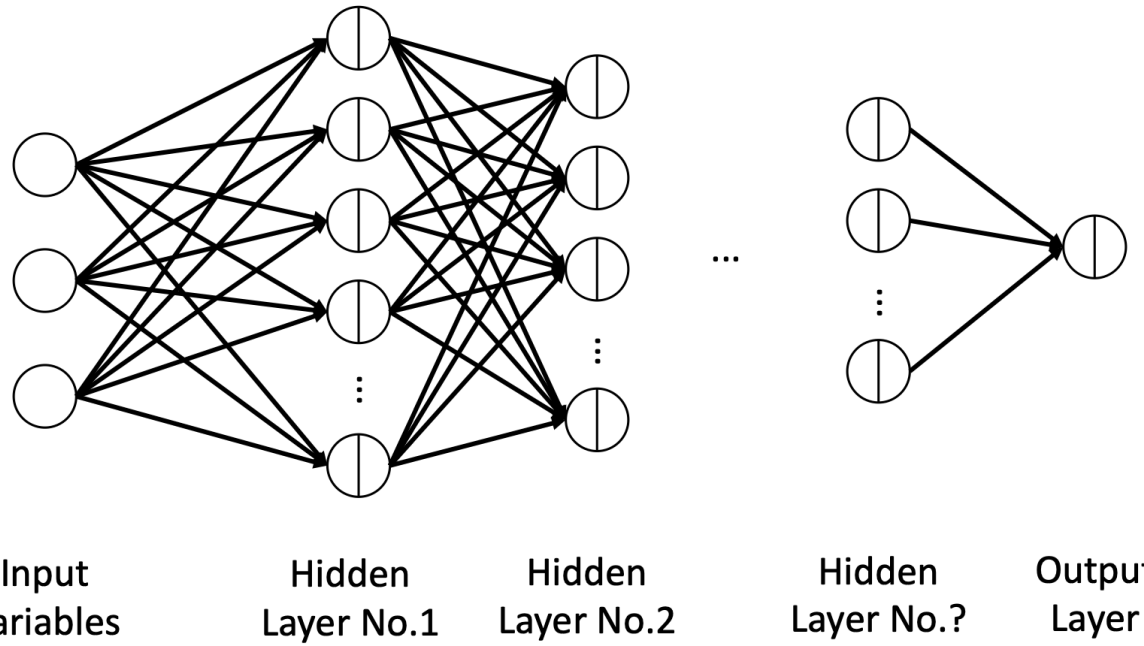
- Logistic Regression Revisited:
  - When $x \in \mathbb{R}^D$ and y $\in \mathbb{R}^1$:

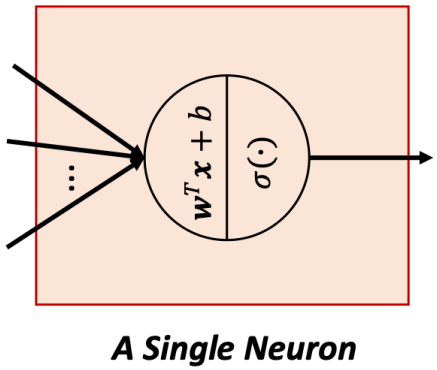
$$y = \sigma(\mathbf{w}^\top \mathbf{x} + b)$$



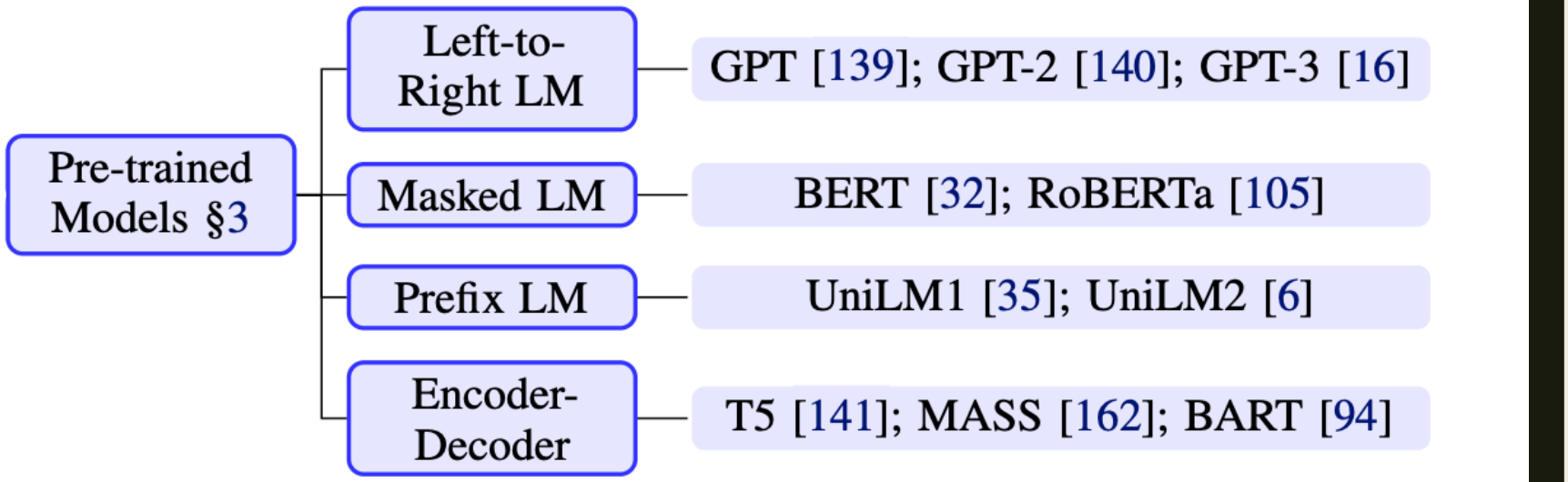
Multi-Layer Perceptron (MLP)

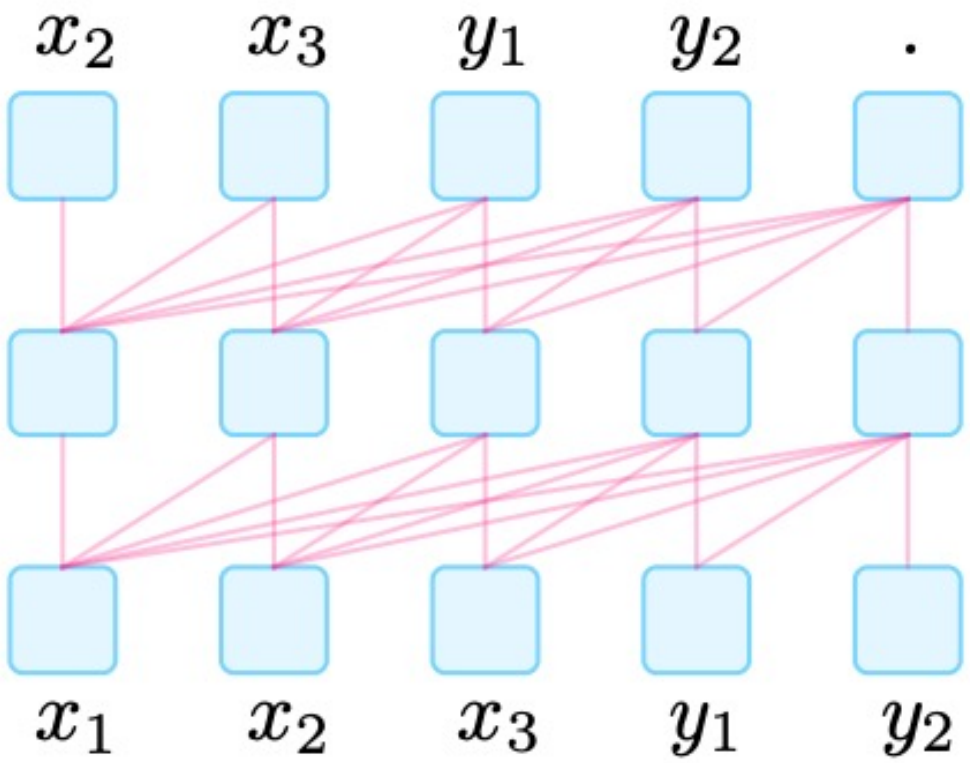
- By stacking the layers, we have ***Multi-Layer Perceptron***:



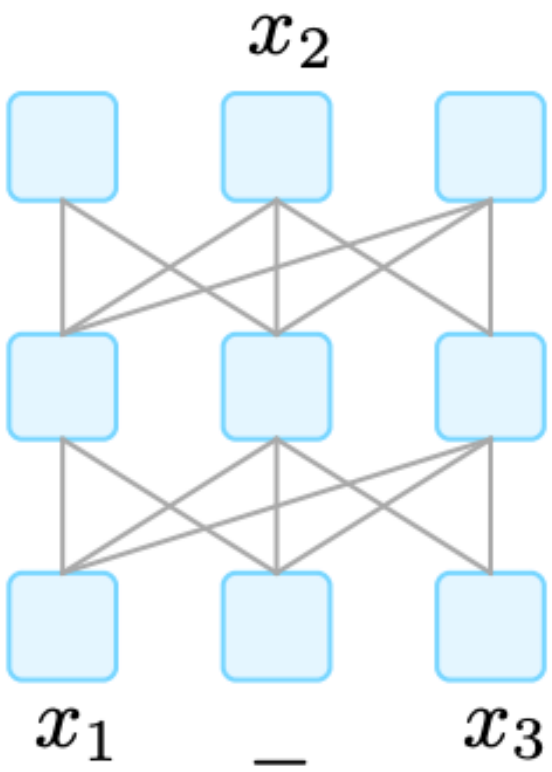


A Single Neuron

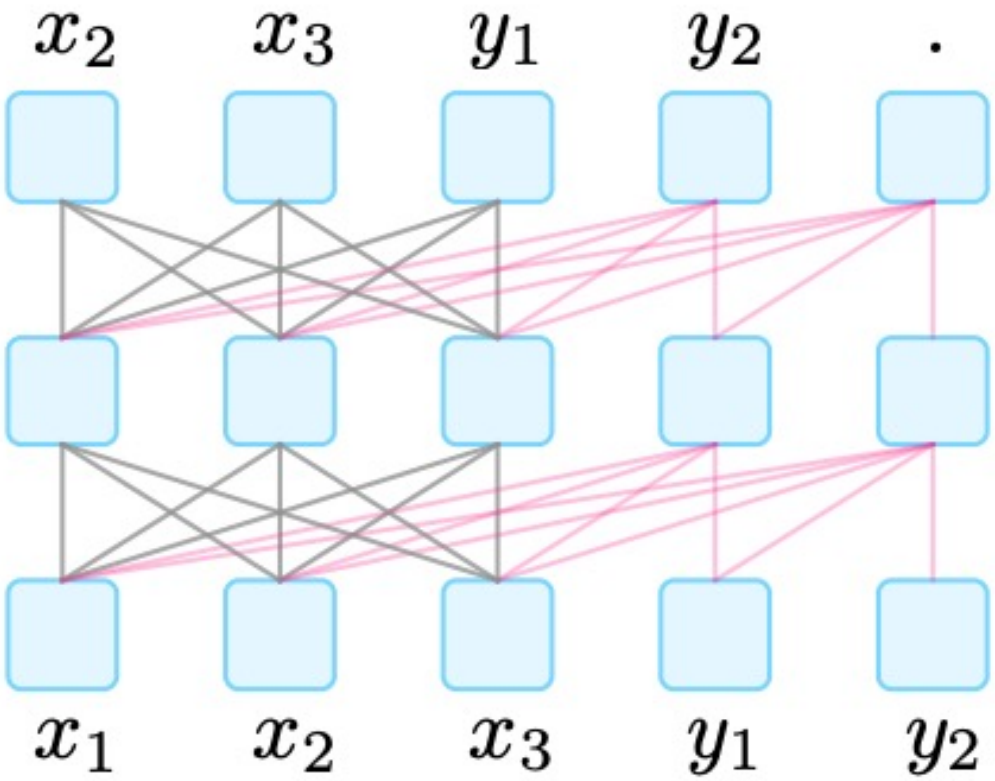

Pre-trained Models §3
Left-to-Right LM
GPT [139]; GPT-2 [140]; GPT-3 [16]
Masked LM
BERT [32]; RoBERTa [105]
Prefix LM
UniLM1 [35]; UniLM2 [6]
Encoder-Decoder
T5 [141]; MASS [162]; BART [94]



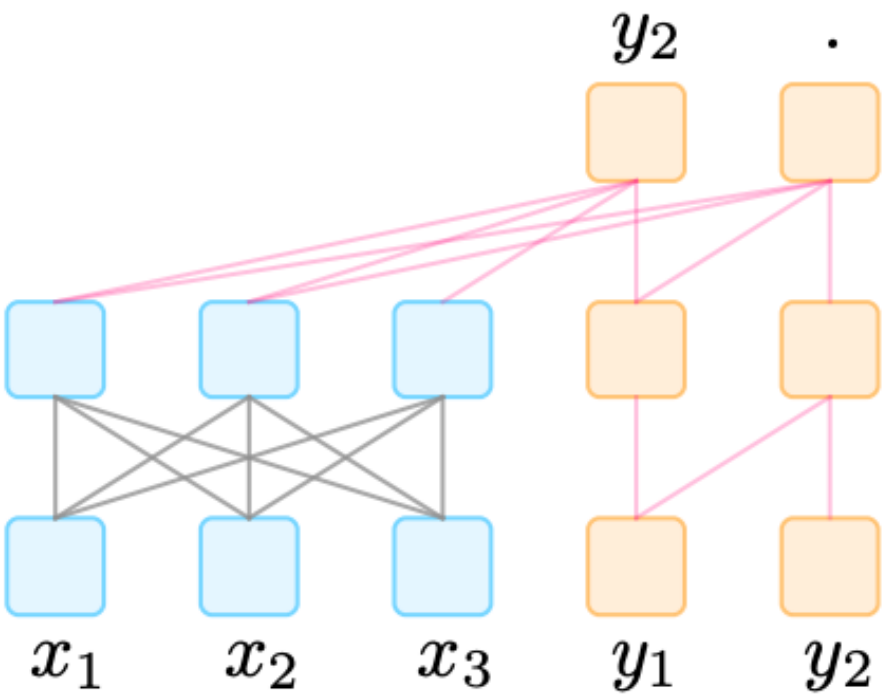
(a) Left-to-right LM.



(b) Masked LM.



(c) Prefix LM.



(d) Encoder-Decoder.

Reference

- https://arxiv.org/pdf/2107.13586.pdf
- https://arxiv.org/pdf/1810.04805.pdf
- https://arxiv.org/pdf/2302.13971.pdf
- https://arxiv.org/pdf/1910.13461.pdf