

# Introduction aux Transformers et LLMs



Aissam Outchakoucht

[a.outchakoucht@emsi-edu.ma](mailto:a.outchakoucht@emsi-edu.ma)

[aissam.outchakoucht@gmail.com](mailto:aissam.outchakoucht@gmail.com)

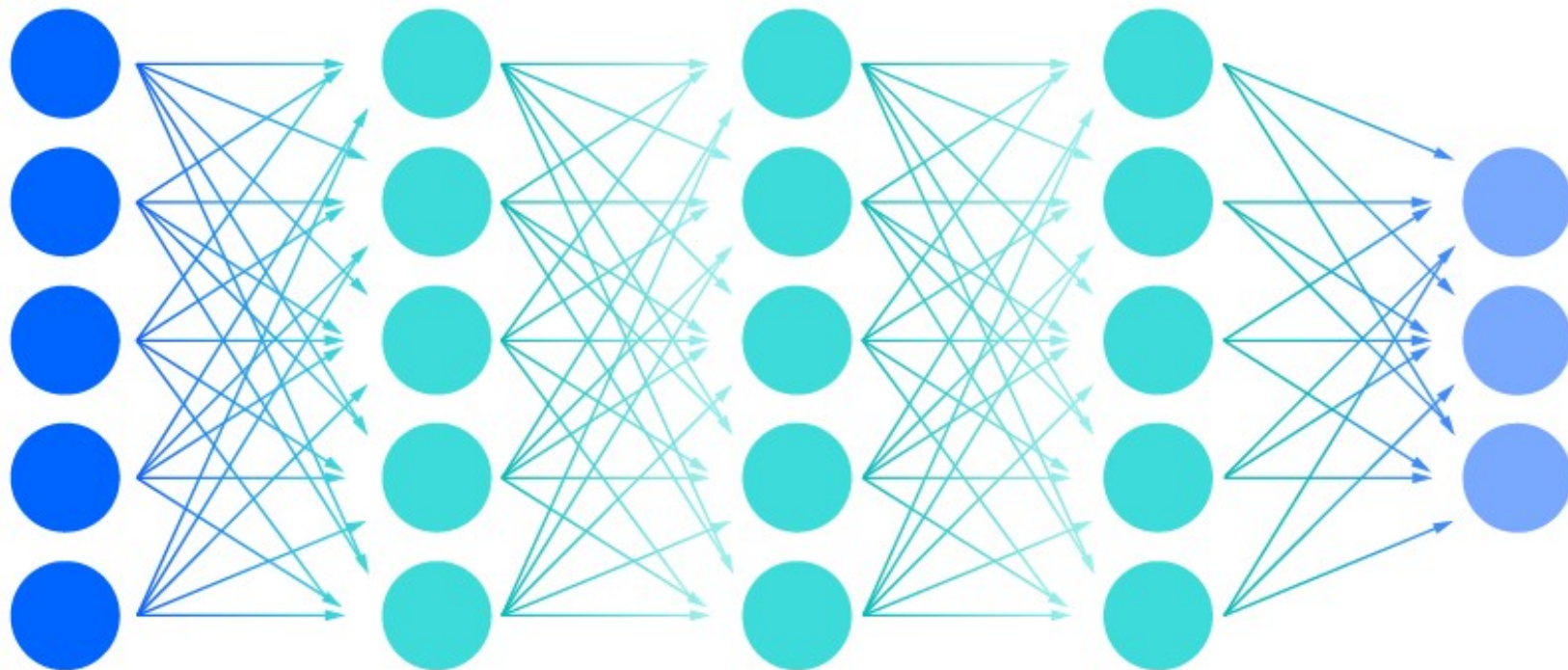


# Recap

Input layer

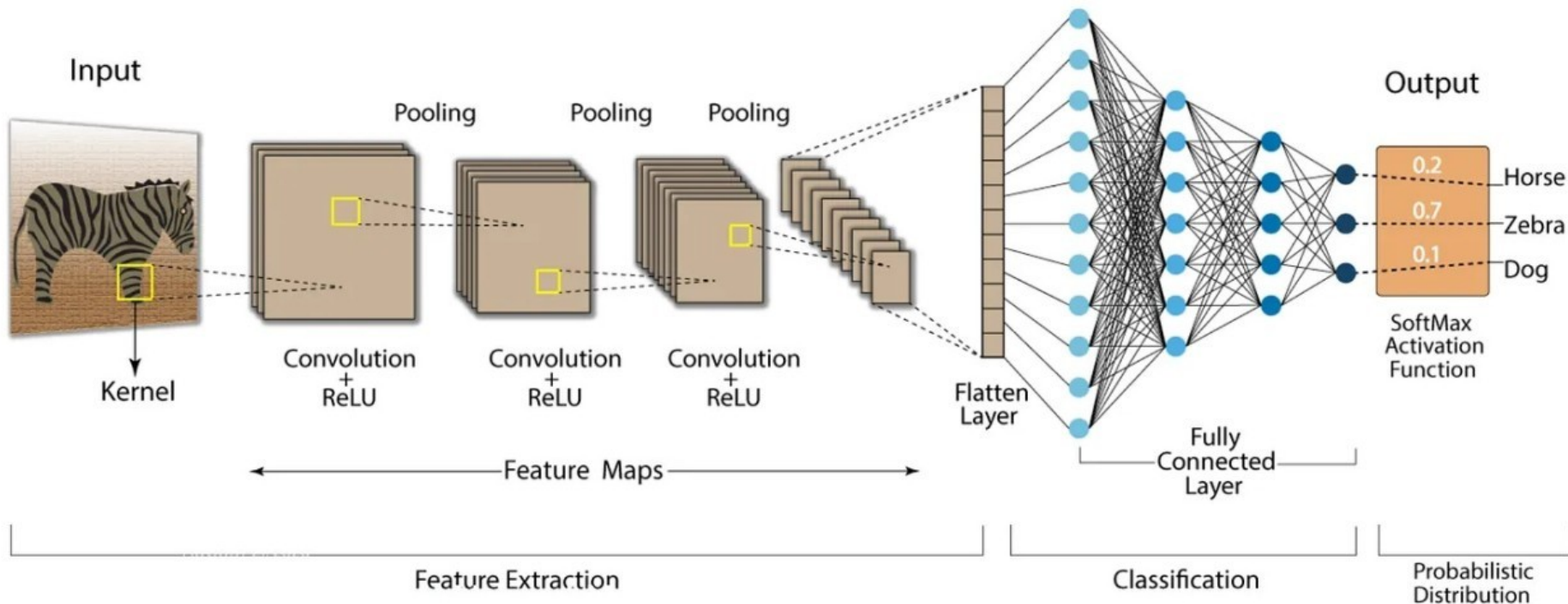
Multiple hidden layer

Output layer

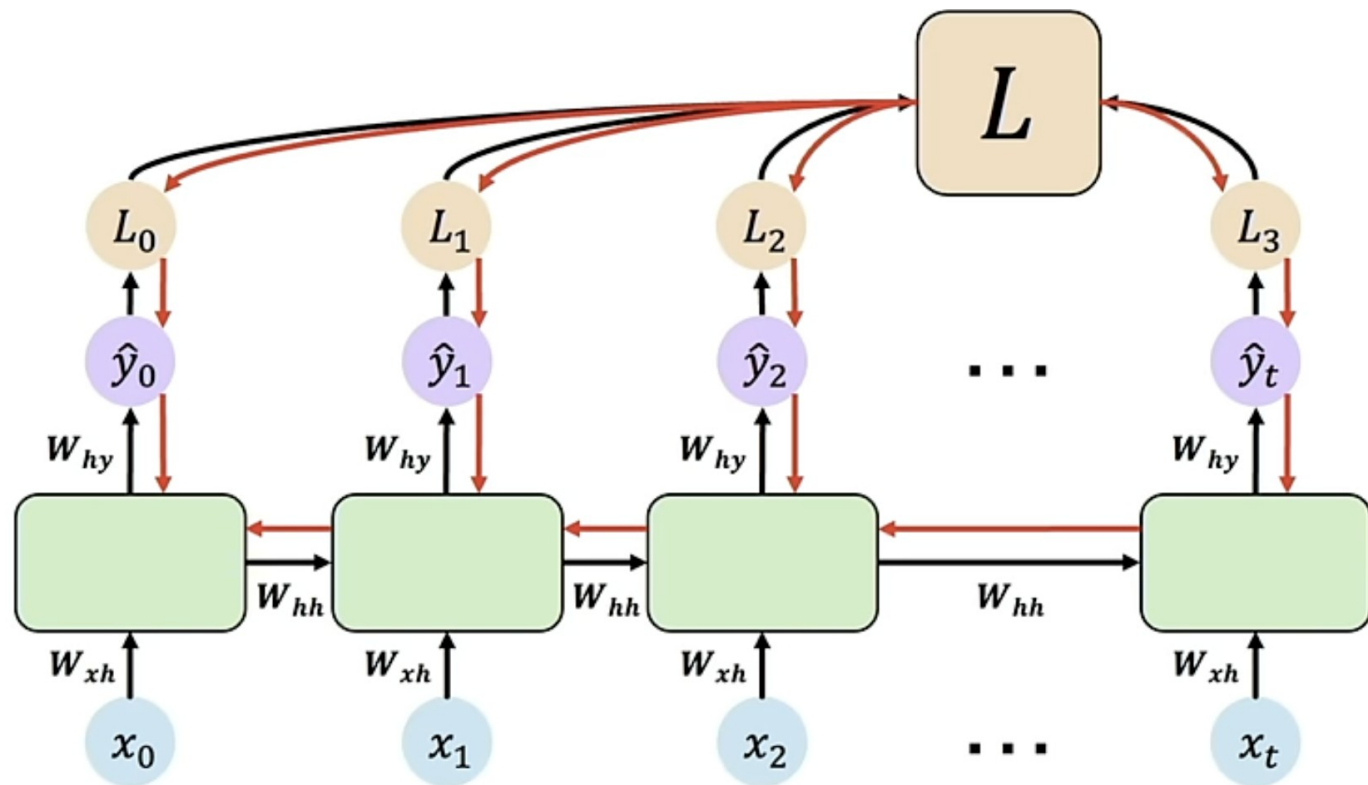


# Recap: CNN

## Convolution Neural Network (CNN)



# Recap: RNN



# Tokenization

hello world! this is a tokenization example

[24912, 2375, 0, 495, 382, 261, 6602, 2860, 4994]

```
from transformers import AutoTokenizer
```

```
tokenizer = AutoTokenizer.from_pretrained("bert-base-uncased")  
tokens = tokenizer.tokenize("I love playing tennis.")
```

# Tokenization

hello world! this is a tokenization example

[24912, 2375, 0, 495, 382, 261, 6602, 2860, 4994]

```
from transformers import AutoTokenizer

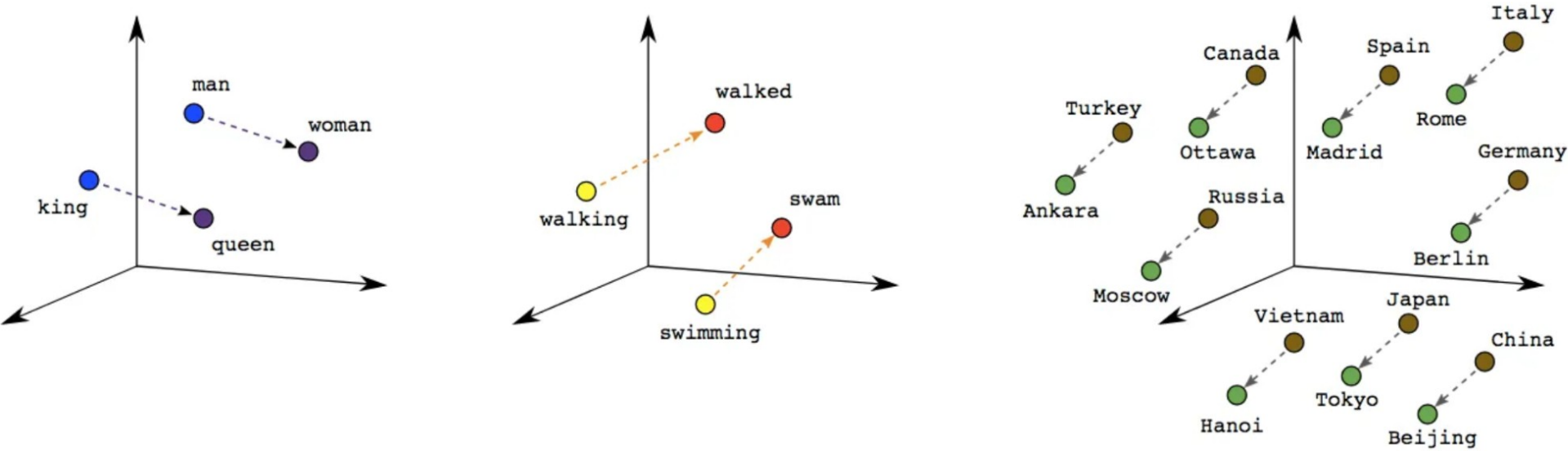
tokenizer = AutoTokenizer.from_pretrained("bert-base-uncased")
tokens = tokenizer.tokenize("I love playing tennis.")
```

La tokenization est le processus de découpage d'un texte en unités plus petites appelées tokens.

# Tokenization

| Caractéristique             | Lettres     | Mots    | Sous-mots  |
|-----------------------------|-------------|---------|------------|
| Taille du vocabulaire       | Petite      | Grande  | Moyenne    |
| Longueur des séquences      | Longue      | Courte  | Moyenne    |
| Gestion des mots OOV        | Excellente  | Faible  | Très bonne |
| Contexte sémantique         | Faible      | Fort    | Moyen      |
| Simplicité du prétraitement | Très simple | Moyenne | Complexe   |

# EMbedding



Un embedding est une représentation vectorielle d'un mot ou d'un token dans un espace multidimensionnel. Il permet de transformer chaque mot/token en un vecteur de nombres réels qui capture ses propriétés sémantiques et contextuelles.

|     |
|-----|
| 3.5 |
| 2.0 |
| 0.2 |
| .   |
| .   |
| .   |
|     |

# Next word prediction

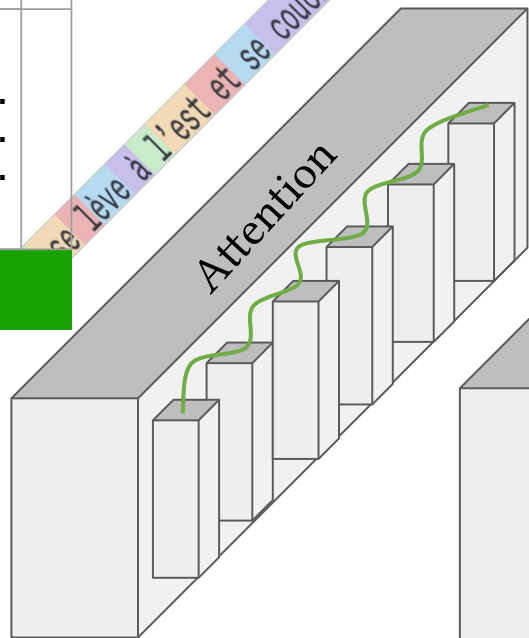
Le soleil se lève à l'est et se couche à ?

...

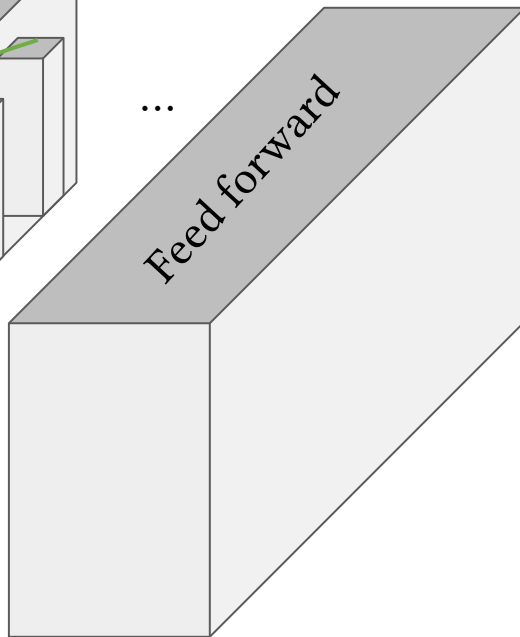
|       |
|-------|
| 0.001 |
| 0.87  |
| 2.0.1 |
| ...   |
| ...   |
| ...   |

# Transformers

Le se lève à l'est et se couche à l'ouest



...



len=vocab size

softmax

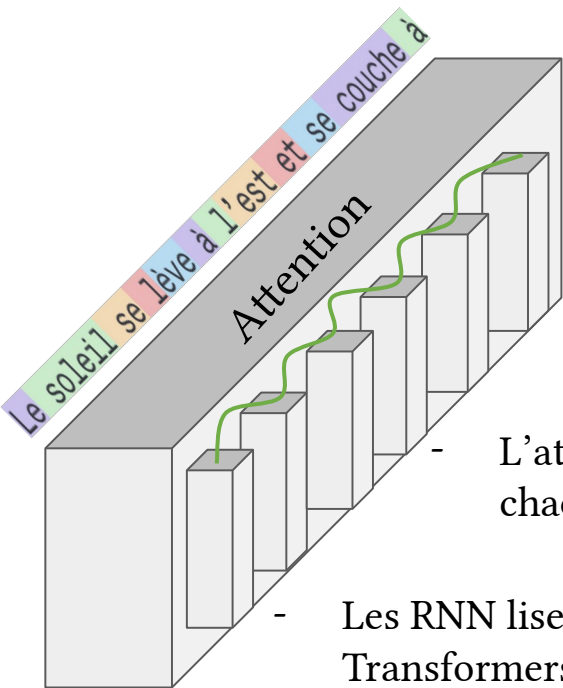
Lookup

l'ouest

$\frac{1}{3}$  of the params

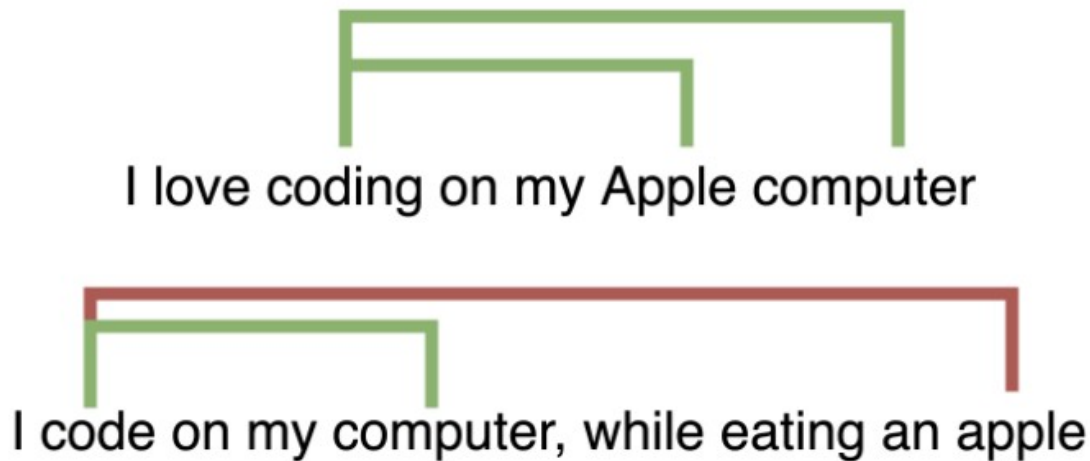
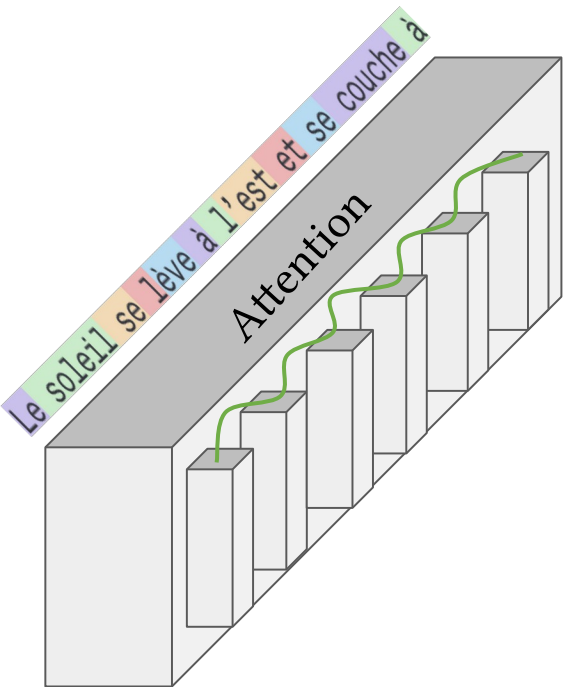
$\frac{2}{3}$  of the params

# Transformers: attention



- Dans une phrase longue, certains mots sont plus importants que d'autres pour comprendre le contexte.
- Les modèles traditionnels (comme les RNN) ont du mal à se souvenir d'informations éloignées dans une séquence.
- L'attention permet au modèle de se concentrer sur les mots importants pour chaque mot de la phrase, même s'ils sont éloignés.
- Les RNN lisent mot par mot et oublient souvent les mots lointains, alors que les Transformers analysent tous les mots en même temps grâce à l'attention.

# Transformers: attention

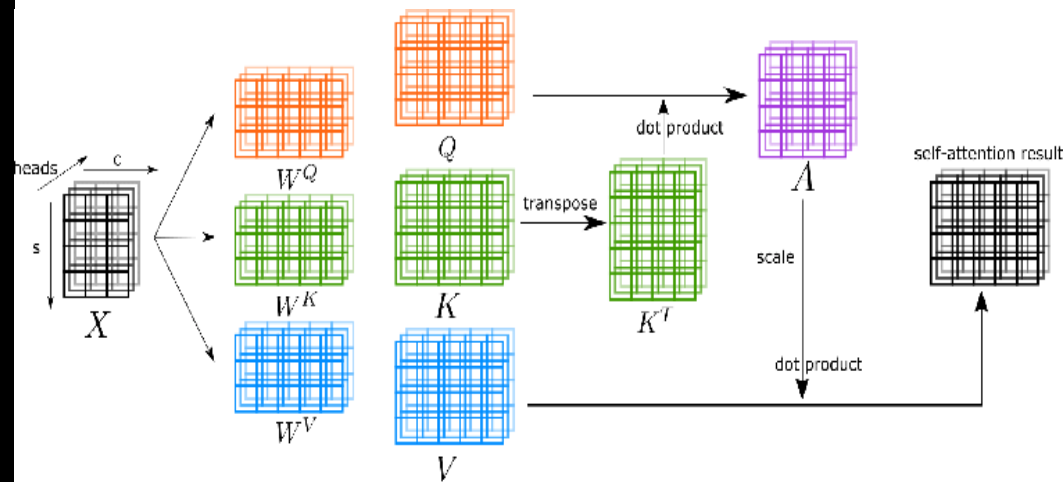


"Each of you has what on the surface appears an unassailable alibi." she said, as the storm cleared and the crimson rays of dusk poured through the window where everyone was gathered. "But only one of you could have known about the loose screw on the window, while also having known where the second key was hidden. I am left, inescapably then, to the conclusion that **therefore, the murderer was** ???



+2.2  
-6.2  
+5.1  
-8.3  
+8.8  
-0.6  
-3.2  
-7.6  
-1.3  
+3.2  
⋮  
-8.0

Le mécanisme d'attention est une technique utilisée dans les modèles de langage pour aider à se concentrer sur les parties importantes d'une phrase. Lorsqu'un mot est analysé, l'attention permet au modèle de "regarder" les autres mots pertinents, même s'ils sont éloignés dans la phrase. Cela aide à mieux comprendre le contexte et les relations entre les mots.



"Each of you has what on the surface appears an unassailable alibi." she said, as the storm cleared and the crimson rays of dusk poured through the window where everyone was gathered. "But only one of you could have known about the loose screw on the window, while also having known where the second key was hidden. I am left, inescapably then, to the conclusion that therefore, the murderer was ???



+2.2  
-6.2  
+5.1  
-8.3  
+8.8  
-0.6  
-3.2  
-7.6  
-1.3  
+3.2  
⋮  
-8.0

Le mécanisme d'attention est une technique utilisée dans les modèles de langage pour aider à se concentrer sur les parties importantes d'une phrase. Lorsqu'un mot est analysé, l'attention permet au modèle de "regarder" les autres mots pertinents, même s'ils sont éloignés dans la phrase. Cela aide à mieux comprendre le contexte et les relations entre les mots.

Query (Q) : Chaque mot pose une question pour savoir quels autres mots sont importants pour lui.

Key (K) : Chaque mot donne des indices pour montrer son importance.

Value (V) : Chaque mot apporte son contenu si on le considère important.

# Transformers: Feed Forward

Le feed forward agit comme une "mémoire" où le modèle stocke et traite les connaissances générales ou "faits" appris pendant l'entraînement.

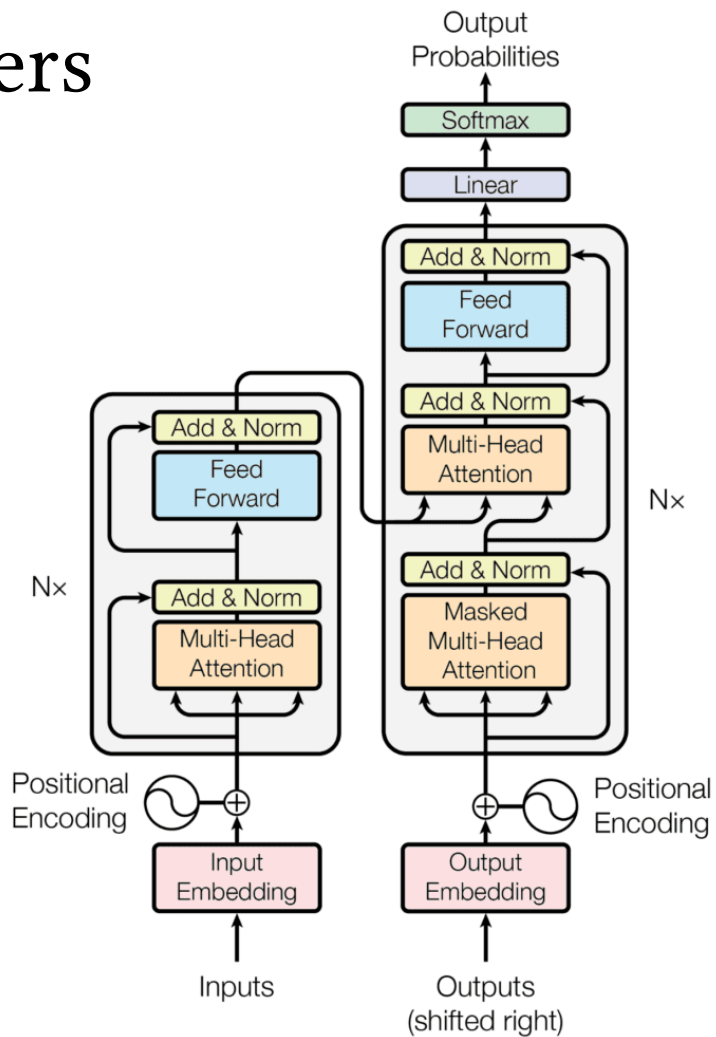
Il transforme la sortie du mécanisme d'attention en une représentation plus utile, combinant les relations contextuelles captées par l'attention avec des modèles plus complexes.

Contrairement à l'attention qui traite les interactions entre les tokens, le feed forward fonctionne indépendamment sur chaque token, permettant d'affiner leur représentation individuelle.

Il affine les vecteurs d'entrée des tokens (issus de l'attention) en extrayant des caractéristiques plus abstraites et riches grâce à ses couches non-linéaires et ses activations (comme ReLU ou GELU).

Il projette les représentations des tokens dans un espace de dimension plus élevée (par exemple, un facteur de 4x la dimension d'entrée), ce qui permet d'apprendre des motifs plus abstraits. Ensuite, il réduit la dimension pour ramener les représentations à la taille d'origine.

# Transformers



# Context length

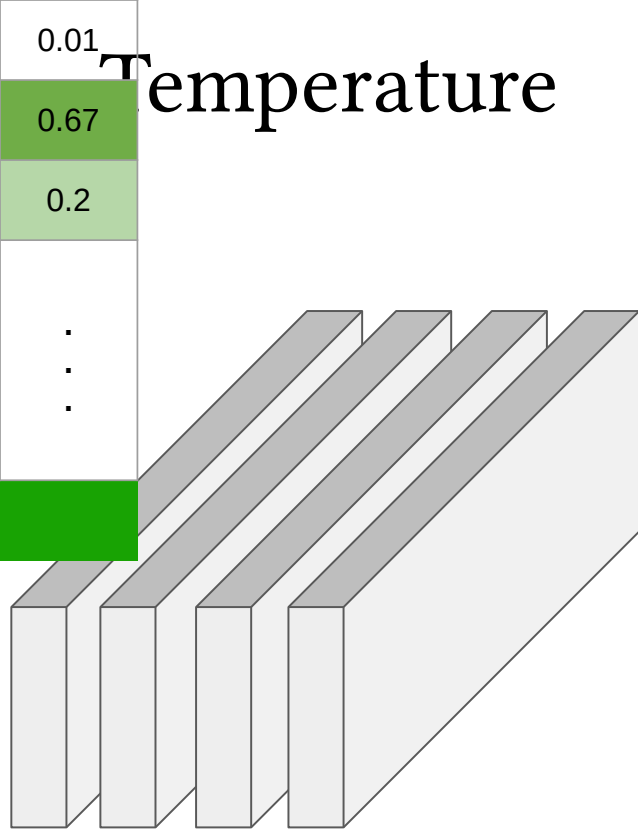
| Model     | Context length | Number of English pages* |
|-----------|----------------|--------------------------|
| GPT 3.5   | 4,096          | 6                        |
| GPT 4     | 8,192          | 12                       |
| GPT 4-32k | 32,768         | 49                       |
| Llama 1   | 2,048          | 3                        |
| Llama 2   | 4,096          | 6                        |

Context length comparison. (\*Assuming 500 words per page.)

C'est le nombre maximum de mots ou tokens qu'un modèle peut traiter en même temps.

Avec un contexte limité, le modèle oublie les informations des débuts de texte lorsqu'il atteint la limite.

Augmenter la longueur du contexte rend les modèles plus lents et demande plus de mémoire.



Un paramètre qui contrôle “la créativité” du modèle en ajustant la distribution de probabilité des mots générés.

Température basse : Génère des sorties plus précises et prévisibles en privilégiant les mots les plus probables.

Température élevée : Favorise des sorties plus variées et créatives en explorant des mots moins probables.