RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

METEO FRANCE
À VOS CÔTÉS, DANS UN CLIMAT QUI CHANGE

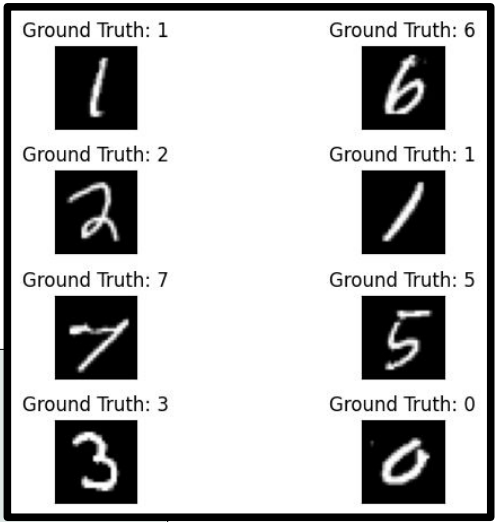
Retour sur le TP n°3 (entraînement d'un CNN)
Présentation du TP n°4 (Transformers)

Pierre Lepetit
ENM, le 7/11/2025

Retour sur le TP 3 :

Nous avons vu comment entraîner un CNN à la classification...

- ...sur un problème de reconnaissance de chiffres manuscrits (I)
  - Construction d'un « Vanilla CNN »
  - Meilleur scores qu'un MLP

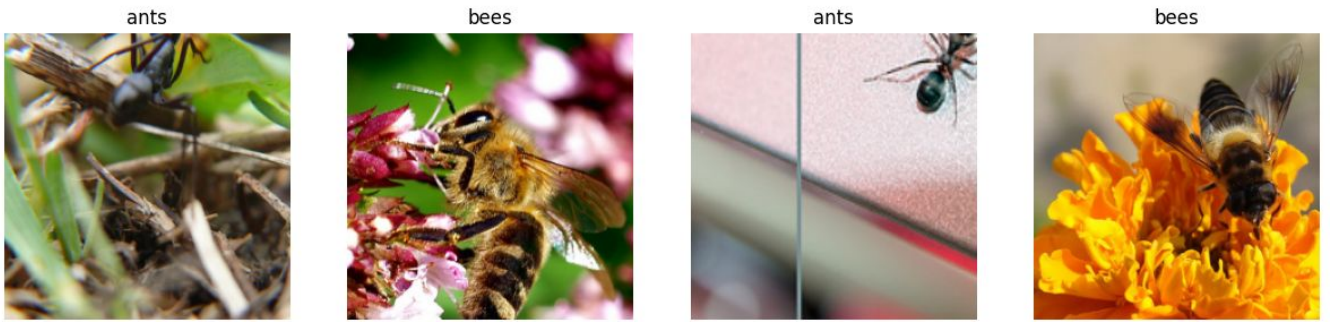



```
class CNN(nn.Module): # Vanilla CNN

    def __init__(self):
        super(CNN, self).__init__()
        self.conv1 = nn.Conv2d(1, 10, kernel_size=5, padding=2)
        self.conv2 = nn.Conv2d(10, 10, kernel_size=5, padding=2)
        self.fc1 = nn.Linear(N, 50)
        self.fc2 = nn.Linear(50, 10)

    def forward(self, x):
        x = F.relu(F.max_pool2d(self.conv1(x), 2))
        x = F.relu(F.max_pool2d(self.conv2(x), 2))
        x = x.view(-1, N)
        x = F.relu(self.fc1(x))
        x = F.dropout(x, training=self.training)
        x = self.fc2(x)
        return F.log_softmax(x, dim=1)
```

Retour sur le TP 3 :

Nous avons vu comment entraîner un CNN à la classification...

- ...sur un problème de reconnaissance de chiffres manuscrits (I)
  - Construction d'un « Vanilla CNN »
  - Meilleur scores qu'un MLP
- ...sur un problème de classification binaire (Hymenoptera dataset)
  - Paramétrage de l'augmentation de données (II.A)



Retour sur le TP 3 :

Nous avons vu comment entraîner un CNN à la classification...

- ...sur un problème de reconnaissance de chiffres manuscrits (I)
  - Construction d'un « Vanilla CNN »
  - Meilleur scores qu'un MLP
- ...sur un problème de classification binaire (Hymenoptera dataset)
  - Paramétrage de l'augmentation de données (II.A)
  - Modification d'un réseau sur étagère (bib pytorch) pré-entraîné(II.B)

```
model = models.resnet18(pretrained=True)
num_ftrs = model.fc.in_features
model.fc = nn.Linear(num_ftrs, 2)
```

Retour sur le TP 3 :

Nous avons vu comment entraîner un CNN à la classification...

- ...sur un problème de reconnaissance de chiffres manuscrits (I)
  - Construction d'un « Vanilla CNN »
  - Meilleur scores qu'un MLP
- ...sur un problème de classification binaire (Hymenoptera dataset)
  - Paramétrage de l'augmentation de données (II.A)
  - Modification d'un réseau sur étagère (bib pytorch) pré-entraîné(II.B)
  - Passage de la descente de gradient sur GPU (II.C) → speed-up 20x

```
model = get_model(pretrained=False).to(device)
optimizer = torch.optim.SGD(model.parameters(), lr=0.001)

for inputs, labels in dataloaders[phase]:
    inputs = inputs.to(device)
    labels = labels.to(device)
    ...
```

Retour sur le TP 3 :

Nous avons vu comment entraîner un CNN à la classification...

- ...sur un problème de reconnaissance de chiffres manuscrits (I)
  - Construction d'un « Vanilla CNN »
  - Meilleur scores qu'un MLP
- ...sur un problème de classification binaire (Hymenoptera dataset)
  - Paramétrage de l'augmentation de données (II.A)
  - Modification d'un réseau sur étagère (bib pytorch) pré-entraîné(II.B)
  - Passage de la descente de gradient sur GPU (II.C)
  - Comparaison entre deux méthodes de Fine-Tuning (II.D, avec ou sans « Freezing »)

```
# resnet_scratch = get_model(pretrained=False)

# Fine-tuning :
resnet_ft = get_model(pretrained=True)

# Gel des poids (freezing):
for module in resnet.modules() :
  if isinstance(module, nn.Conv2d) or isinstance(module, nn.BatchNorm2d):
    for param in module.parameters():
      param.requires_grad = False
```

II) Les Transformers

Rapide historique

- 1950s – 1990s : NLP symbolique, modèles statistiques (e.g. : n-grams)
- 2001 : Neural Language Models (Bengio et al.) → neural embedding
- 2013 - 2015 : Word2Vec, RNNs (eg : seq2seq for translation) + Attention mecanisms
- 2017 : Attention is all you need (transformers)
- 2018 : BERT → bidirectional masking ; breakthrough in how to fine-tune
- 2018 - 2019 : GPT1-2 autoregressive transformer + scaling laws
- 2020 : T5, BART, ELECTRA, ViT (ViT : first Vision Transformer)
- 2021 – 2022 : sparse/efficient Attention (Longformer, Performer, FlashAttention etc)

II) Les Transformers

Ce dont on parlera :

- 1950s – 1990s : NLP symbolique, modèles statistiques (e.g. : n-grams)
- 2001 : Neural Language Models (Bengio et al.) → neural embedding
- 2013 - 2015 : Word2Vec, RNNs (eg : seq2seq for translation) + Attention mecanisms
- 2017 : Attention is all you need (transformers)
- 2018 : BERT → biderectional masking ; breakthrough in how to fine-tune
- 2018 - 2019 : GPT1-2 autoregressive transformer + scaling laws
- 2020 : T5, BART, ELECTRA, ViT (ViT : first Vision Transformer)
- 2021 – 2022 : sparse/efficient Attention (Longformer, Performer, FlashAttention etc)

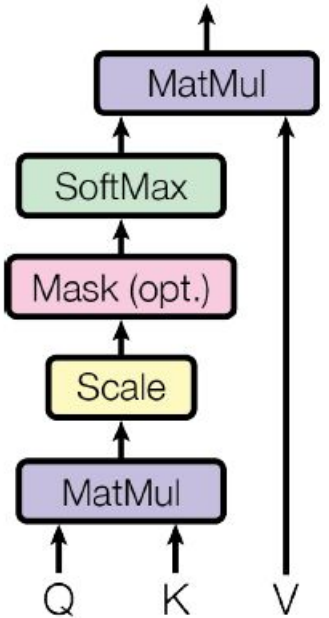
II) Les Transformers

Tokenisation et couche attentionelle

- **Idée 1** : input = suite finie de L tokens
  Exemples:
  NLP : standardisation du prompt + segmentation (e.g. BPE)
  CV : découpe en patchs et applatissement

- **Idée 2** : Mapping token → L tenseurs Xi
  Exemples :
  NLP : token id → vecteur entraînable de taille D_model
  CV : 3x16x16 patches → conv2d(stride=16)
  → vecteurs de taille D_model

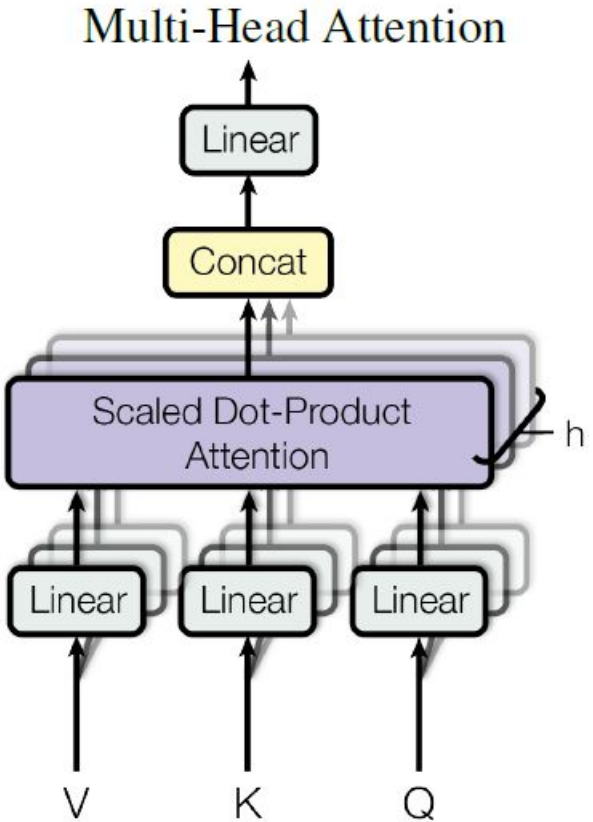
- **Idée 3** : « attention = produit scalaire » en trois phases :
  Projection : $Q_i = X_i W_q$, $K_i = X_i W_k$, $V_i = X_i W_v$
  Score d'attention : $S = Q_i \times {}^tK_i$ (+norm.)
  Moyenne pondérée des « values » : $Xi := \Sigma_k S_{ik} V_k$



II) Les Transformers

MultiHead Layers :

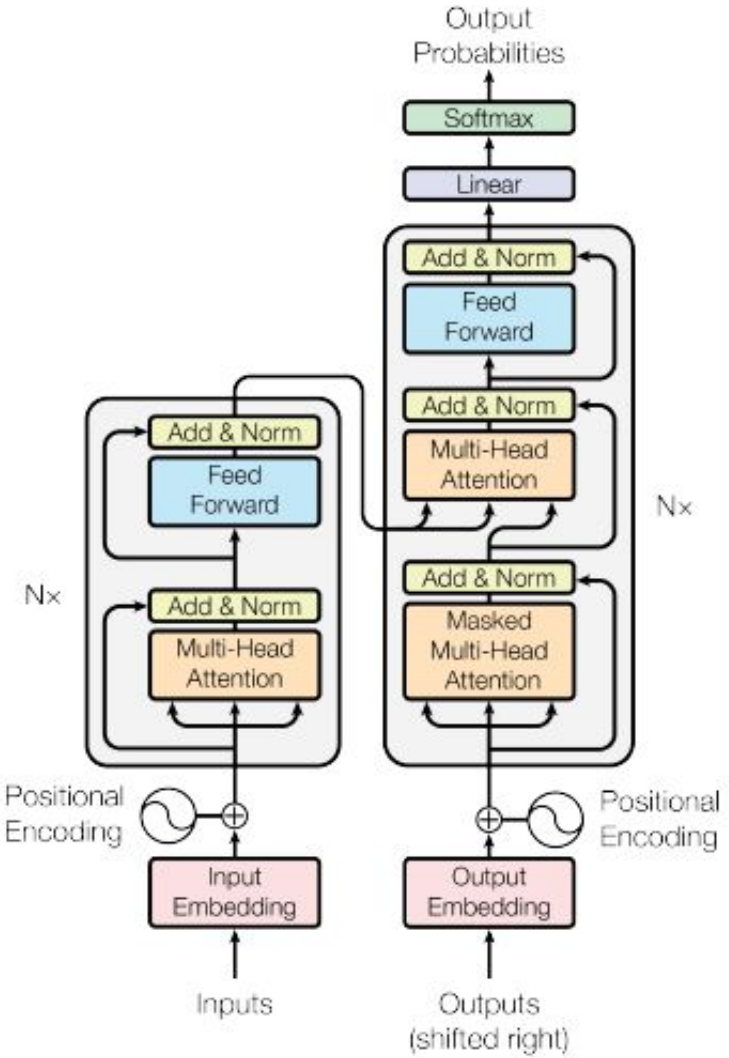
- N_h tête attentionnelles = N_h matrices W_q, W_k, W_v = N_h calculs d'attention
- Intérêt :
  - « spécialisation » des têtes
  - Parallélisation des claculs d'attention
- En pratique :
  - Ajout d'une dimension supplémentaire dans les tenseurs
  - Résduction de la taille à la sortie (concaténation)



II) Les Transformers

Architecture globale pour de la traduction :

- Inputs & targets ?
- Mode autorégressif (+ matrice trian
- Rôle de l'encodeur
- Couche spécifique (attention croisé
- Positional encoding



II) Les Transformers

Présentation du TP n°4 :

- Visualisation des Q, K, V
- Multihead attentional layers
- Présentation d'exemples jouets :
  - Faciles à simuler
  - Pour lesquels un « vanilla CNN » << Transformer
- Effet de la dilation (?)