

Large language Models (ADIA94)

Département : Big Data & Machine Learning

Disciplines : Sciences des données et intelligence artificielle

Enseignant : Stefani EL KALAMOUNI - stefani.el-kalamouni@efrei.fr

TP4 - Exercice 1 : Implémentation du mécanisme de self-attention

Objectif

Comprendre et implémenter le mécanisme de self-attention utilisé dans les Transformeurs. À partir d'une phrase donnée, vous devrez créer des vecteurs pour chaque mot, calculer les scores d'attention, et produire les vecteurs finaux pondérés.

Astuces

- Utilisez la bibliothèque `numpy` pour les calculs matriciels.
- Utilisez la fonction `np.exp` pour exponentier les scores et `np.sum` pour calculer la somme dans la softmax.
- Vérifiez les dimensions de vos matrices à chaque étape pour éviter les erreurs

1. Créez des embeddings pour chaque mot

- Représentez chaque mot d'une phrase donnée par un vecteur d'embedding.
- Par simplicité, créez des embeddings aléatoires de petite dimension ($d_{\text{embedding}}=4$).
- Exemple de phrase : "**Le chat noir dort.**"

2. Calculez les vecteurs Query, Key, et Value

- Les vecteurs Query Q , Key K , et Value V sont obtenus en multipliant les embeddings des mots par des matrices de poids (W_Q , W_K , W_V).
- Les dimensions des matrices de poids doivent être compatibles :
 - Si les embeddings ont une dimension $d_{\text{embedding}}$, les matrices de poids doivent avoir une dimension $d_{\text{embedding}} \times d_{\text{head}}$ où d_{head} est la dimension des vecteurs Q , K et V

3. Calculez les scores d'attention

- Les scores d'attention sont calculés en effectuant un produit scalaire entre les vecteurs Query et Key pour chaque paire de mots.

Formule mathématique :

$$\text{Score}_{ij} = Q_i \cdot K_j^T$$

où Q_i est le Query du mot i et K_j est le Key du mot j .

4. Appliquez la softmax

- Les scores d'attention sont transformés en probabilités à l'aide de la softmax
- Cela garantit que les pondérations sont comprises entre 0 et 1, et que leur somme pour chaque mot est égale à 1.

5. Obtenez la sortie pondérée

- Multipliez les pondérations (issues de la softmax) par les vecteurs Value (VVV) pour obtenir la sortie pondérée pour chaque mot.
- Formule mathématique :

$$\text{Output}_i = \sum_j \text{AttentionWeight}_{ij} \cdot V_j$$

6. Affichez les résultats

- Affichez :
 - Les pondérations (matrice $n_{\text{words}} \times n_{\text{words}}$)
 - Les vecteurs finaux pour chaque mot.

Consignes

1. **Phrase à traiter** : "Le chat noir dort."
2. **Dimension des embeddings** : $d_{\text{embedding}}=4$
3. **Dimension des vecteurs Q,K,V** : $d_{\text{head}}=2$

Exercice 2 : Mise en œuvre complète d'une attention multi-têtes avec encodage de position

Objectif

1. Implémenter un mécanisme de **self-attention multi-têtes**.
2. Ajouter un **encodage de position** aux embeddings des mots.
3. Visualiser les **pondérations d'attention** pour mieux comprendre les relations entre les mots dans une phrase.

Consignes

1. **Phrase d'exemple** : Travaillez sur la phrase : "**Le soleil brille doucement sur la rivière.**"
2. **Caractéristiques des embeddings** :
 - Chaque mot sera représenté par un embedding de dimension $d_{\text{embedding}}=6$.
 - Implémentez **2 têtes d'attention**, avec $d_{\text{head}}=3$ pour chaque tête.
3. **Étapes à réaliser** :
 - a. **Encodage de position** :
 - Calculez et ajoutez des vecteurs d'encodage de position à chaque embedding.
 - Utilisez des fonctions sinus et cosinus pour l'encodage (comme expliqué en cours).
 - b. **Mécanisme de self-attention multi-têtes** :
 - Implémentez la **self-attention** pour chaque tête.
 - Combinez les résultats des têtes en **concaténant leurs sorties**.
 - c. **Pondération et sortie finale** :
 - Ajoutez une couche linéaire finale pour obtenir un vecteur de dimension $d_{\text{model}}=6$.
4. **Visualisation** :
 - Affichez les **scores d'attention** pour chaque mot (matrice $n_{\text{words}} \times n_{\text{words}}$).
 - **Interprétez les relations entre les mots** basées sur les scores.
5. **Bonus (facultatif)** :
 - Essayez de **modifier la phrase** (par exemple, en remplaçant "soleil" par "lune") et observez comment les pondérations changent.