

Report sulle differenze tra i due codici

Entrambi i codici implementano un modello U-Net per la segmentazione delle immagini utilizzando TensorFlow/Keras.

Tuttavia, differiscono in alcuni aspetti chiave relativi alla gestione delle metriche e alla visualizzazione dei risultati.

Ecco un'analisi approfondita delle differenze e di come i loro risultati siano complementari.

Differenze principali

1. Gestione delle metriche durante l'addestramento:

- **Primo codice:**
 - Le metriche IoU e DICE vengono monitorate **durante l'addestramento** grazie a un callback personalizzato (MetricsCallback). Ciò consente di tracciare l'evoluzione di queste metriche a ogni epoca, fornendo un feedback continuo sulle prestazioni.
 - **Secondo codice:**
 - Le metriche IoU (Intersection over Union) e DICE vengono calcolate **dopo l'addestramento**, valutando le performance del modello sulle immagini di validazione. Queste metriche non influenzano l'addestramento in tempo reale.
-

2. Monitoraggio e visualizzazione delle metriche:

- **Primo codice:**
 - Le metriche IoU e DICE vengono tracciate per ogni epoca e plottate come grafici globali alla fine dell'addestramento. Questo approccio consente di valutare l'andamento delle metriche e di correlare eventuali fluttuazioni con il training.
- **Secondo codice:**
 - Le metriche IoU e DICE vengono calcolate e visualizzate graficamente per un sottoinsieme casuale di immagini di validazione alla fine dell'addestramento. Le prestazioni sono rappresentate come overlay visivi e grafici delle metriche per immagine.

3. Visualizzazione dei risultati:

- **Primo codice:**

- Mette in evidenza l'andamento globale delle metriche durante l'intero processo di addestramento. Non fornisce una visualizzazione dettagliata immagine per immagine.

- **Secondo codice:**

- Si concentra sulla valutazione dettagliata di un campione casuale di immagini di validazione. Ogni immagine mostra:
 - La maschera vera.
 - La maschera predetta.
 - Un overlay tra le due.
 - Una rappresentazione grafica delle metriche IoU e DICE per immagine.
-

4. Callback personalizzato:

- **Primo codice:**

- Implementa un callback personalizzato per il monitoraggio delle metriche IoU e DICE. Questo approccio automatizza il calcolo delle metriche a ogni epoca.

.

- **Secondo codice:**

- Non utilizza callback personalizzati. Le metriche sono calcolate manualmente dopo l'addestramento

5. Risultati delle metriche:

- **Primo codice:**
 - Fornisce sia i valori medi di IoU e DICE finali che la loro evoluzione nel tempo. Questo è utile per identificare overfitting o problemi di apprendimento.
 - **Secondo codice:**
 - Produce i valori medi di IoU e DICE per l'intero set di validazione, fornendo una valutazione finale delle prestazioni.
-

Complementarità dei risultati

- **Monitoraggio durante l'addestramento (Primo codice):**
 - Offre una visione globale delle prestazioni del modello durante l'addestramento. L'andamento delle metriche per epoca aiuta a diagnosticare problemi come overfitting o underfitting.
 - Questo approccio è cruciale per ottimizzare l'addestramento e scegliere il numero ottimale di epoche.
 - **Approccio post-addestramento (Secondo codice):**
 - È utile per un'analisi approfondita dei risultati finali. La visualizzazione immagine per immagine permette di identificare casi specifici in cui il modello funziona bene o meno.
 - Questo approccio è ideale per la valutazione finale del modello, in particolare se si desidera un feedback visivo dettagliato.
-

Conclusione Metriche

La combinazione di **Dice Coefficient** e **IoU** fornisce una valutazione robusta e completa della performance del modello nella segmentazione binaria. La scelta di queste metriche è motivata dalla natura del problema, ovvero segmentare precisamente le aree di interesse (come polipi o altre anomalie) in immagini mediche per ridurre al minimo gli errori di classificazione.

Il **Dice Coefficient** è particolarmente indicato per questo tipo di applicazioni perché penalizza fortemente le discrepanze nelle piccole aree, che sono molto comuni nelle immagini mediche.

L'**IoU** è una metrica complementare che consente di valutare l'efficienza complessiva del modello nel distinguere correttamente l'area di interesse dalla parte non pertinente dell'immagine. Infine, l'**accuracy** offre una visione generale del comportamento del modello, ma dovrebbe essere integrata con altre metriche per una valutazione completa.

Conclusione

I due approcci sono complementari:

- Il **primo codice** fornisce un monitoraggio continuo e dettagliato delle prestazioni durante il training.
- Il **secondo codice** è orientato verso un'analisi qualitativa e quantitativa dei risultati finali.