**PROGETTO ALGORITMI PER IL WEB**

**Introduzione**

Il seguente progetto si pone l’obiettivo di costruire un classificatore in grado di determinare la razza di un cane data un’immagine. Per la risoluzione del problema è stato utilizzato il metodo del Transfer Learning, tale metodo consente il riutilizzo dei pesi di una rete neurale sviluppata per altri scopi come punto di partenza della propria rete neurale.

Le principali librerie utilizzate sono quelle che implementano gli algoritmi di machine learning: tensorflow, keras.

**Documenti**

Per la costruzione della rete neurale il software sviluppato effettua un load dei dati presenti nella piattaforma kaggle. Tali dati comprendono un documento di immagini non classificate utilizzato per il test, una directory di training contenente sub directory con le immagini dei cani relativi alla razza con cui è denominata la sotto-directory. Infine è presente un documento contenente un’associazione idCane ,razza.

**Sviluppo**

Una volta scaricato il dataset dalla piattaforma di kaggle vengono divise le immagini in un sistema di sottocartelle denominate training set e validation set, contenenti rispettivamente il 75% e 25% del dataset. Queste serviranno per l’addestramento e la validazione del nostro modello.

Per il training set è stata utilizzata la tecnica del Data Augmentation che permette di modificare la forma delle immagini secondo dei criteri scelti per ottenere una maggiore capacità di generalizzazione della rete.

Tramite Keras è stato possibile preprocessare e normalizzare i dati con la funzione ImageDataGenerator.

Per raggiungere il nostro obiettivo è stata utilizzata la tecnica del transfer learning, dove viene preso un modello già addestrato relativo ad un altro problema ed usato come punto di partenza per il nostro.

Il modello pre-trained considerato è quello della rete Nasnet: si è visto dallo studio di progetti in cui è richiesto il riconoscimento di razze di cani che tale rete ha un’alta accuratezza. Sopra la rete già addestrata, sono stati inseriti dei livelli per adattare l’output della rete al numero di classi del nostro caso in esame. A tale scopo sono stati utilizzati due dense layer ai quali si è applicato un dropout gaussiano come livello di regolarizzazione.

Durante la fase di addestramento è stata utilizzata la tecnica dell’early-stopping: viene terminata la fase di training quando in un epoca non ci sono stati dei miglioramenti del valore di accuracy rispetto all’epoca precedente.

**Risultati**

L’accuracy ottenuta sul validation con tale configurazione è di 97.77%

