PEAS

Performance: quanto è positivo? Qualcosa sui pixel?

Environment: l’ambiente dell’agente sarà composto da un’immagine rappresentante la radiografia al torace di un paziente. Parliamo quindi di un ambiente completamente osservabile, a singolo agente, discreto, noto, statico, sequenziale, deterministico

Actuators: mostra a video il risultato

Sensors: immagini

Il problema preso in esame consiste nell’analizzare delle immagini rappresentanti le radiografie al torace di pazienti ospitati nella struttura ospedaliera che ha adottato la piattaforma di gestione integrata del team Hilo. Risulta conveniente analizzare queste radiografie poiché chi è affetto da Covid-19 presenta delle macchie diffuse sulla quasi totalità della superficie polmonare che man mano si sviluppano seguendo i vasi sanguigni presenti in quel luogo; differisce dalla normale polmonite interstiziale poiché in quel caso caso la presenza è localizzata in uno o più punti ben circoscritti all’interno del polmone (si parla di veri e propri focolai). Sfruttando la caratteristica dell’estensione delle suddette macchine è possibile addestrare un algoritmo di intelligenza artificiale per stabilire se un paziente sia affetto da Covid-19 o meno e quindi fornire una stima di quanto possa essere urgente (e utile) effettuare un tampone molecolare, fornendo un’ulteriore metrica di priorità al gestionale. Così facendo si può gestire una coda di tamponi da effettuare in maniera efficiente e coerente con i veri bisogni dei pazienti.

Da questa definizione si evince che ci si trova davanti ad un problema di apprendimento perché lo scopo è quello di addestrare un agente artificiale a riconoscere la presenza di una infezione da Covid-19 all’interno dei polmoni differenziandola da quella che può essere una normale polmonite generica o, banalmente, dall’assenza di infezioni.

Definita quindi la branca del problema, un nodo da sciogliere è rappresentato dalla natura di quest’ultimo. Per “natura del problema” ovviamente si intende se esso consista in un problema di Classificazione, di Regressione o di Clustering. Dal momento che è stato possibile reperire un dataset di radiografie toraciche provviste di etichette che distinguono i vari casi presi in considerazione dal problema (ovvero assenza di polmonite, presenza di polmonite generica e presenza di polmonite da Covid-19) in automatico è possibile escludere il Clustering come strada da intraprendere. È inoltre pacifico stabilire che non ci si trova in una situazione di Regressione in quanto lo scopo non è fornire una percentuale di positività al Covid-19 ma una “risposta secca” del tipo “Covid SI”, “Covid No” e “Covid NO con Polmonite”. Da quest’ultima considerazione (e per esclusione) si può dedurre che ci si trova di fronte ad un problema di Classificazione.

Per l’implementazione si è deciso di optare per una rete neurale in quanto, a differenza di random forest e alberi di decisione, è l’unica struttura che è in grado di lavorare con delle immagini. Adesso però il problema è decidere quale tipologia di rete neurale adottare. In primo luogo, abbiamo pensato ad una rete neurale a singolo livello (percettrone) però, a seguito della lezione a proposito, lavora solo con problemi linearmente separabili e quindi l’abbiamo scartato. Dopodiché abbiamo pensato ad una rete neurale multilivello o MLP (Multi Level Perceptron), ovvero una struttura composta da un livello di input, un numero arbitrario di livelli intermedi chiamati “livelli nascosti” e un livello di output, che è in grado di risolvere anche problemi non linearmente separabili però a seguito di ricerche su internet, e sulla base dei consigli del prof, abbiamo scartato anche questa tipologia poiché nonostante la sua potenza, in particolare con le immagini non lavora bene e quindi non è in grado di fornirci i risultati che ci aspettiamo. Quindi abbiamo infine deciso di utilizzare come struttura di riferimento una rete neurale convoluzionale (CNN o Convolutional Neural Network), che è definita in un modo molto simile alla MLP però con una sostanziale differenza nella strutturazione dei livelli. Nella CNN abbiamo un livello di Convoluzione, un livello ReLu e un livello di pooling. Il livello di convoluzione si basa su un procedimento simile al passaggio di una lente di ingrandimento sull’input in modo da riconoscere componenti particolari via via meno elementari con l’aggiunta di ulteriori livelli di convoluzione e una importante caratteristica di questo livello è che i neuroni all’interno di questo livello condividono sia pesi che bias in input, simulando una visione sovrapposta dell’immagine. Successivamente al livello di convoluzione troviamo di solito un livello ReLu, che ha lo scopo di applicare la funzione ReLu ( ) che ci permetterà di eliminare valori negativi dalla propagazione in avanti in quanto questi possono creare problemi come lo “spegnimento di neuroni” nel processo di backpropagation. Successivamente a questo livello possiamo trovare anche un livello di pooling il quale è responsabile di un ulteriore filtraggio del risultato proveniente dal livello di convoluzione. Questa operazione consiste nel creare un filtro di una determinata dimensione e nell’estrarre i valori (solitamente i massimi) della zona analizzata dal filtro. Il risultato sarà quindi un ulteriore riduzione dell’output però con un minor rischio di portare la rete in overfit. Infine, troviamo il livello di output che è un livello fully connected (ovvero che ha i neuroni collegati completamente a tutti gli output del livello precedente) e ha semplicemente lo scopo di eseguire la classificazione finale.  
Un’altra peculiarità sta nel fatto che non si è vincolati al numero di livelli utilizzabili ma se ne possono usare quanti ne vogliamo e soprattutto l’ordine di posizionamento di questi non è del tutto rilevante, ma solitamente si preferisce impostare un ordinamento del tipo Conv, ReLu, Pool, Conv, ecc. .

//appunti

MLP vs CNN (Multi-Layer Perceptron vs Convolutional Neural Network)

Una sostanziale differenza tra I due è che nel caso di una MLP, i livelli nascosti sono sempre completamente collegati con gli altri livelli (specialmente con l’output), mentre nel caso di una CNN i livelli non vengono collegati completamente ma ci sono dei processori elementari che condividono il peso dell’input con altri?