



UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

CORSO DI LAUREA TRIENNALE IN

INGEGNERIA INFORMATICA

# Classificazione e rappresentazione di enzimi tramite 3D CNN

*Relatore:*

PROF. NANNI LORIS

*Laureando:*

BRUGNERA ALESSANDRO

1190178

Anno Accademico 2021/2022

**Sommario**

Le reti neurali convoluzionali sono delle reti artificiali che attualmente rappresentano uno dei più usati algoritmi di deep learning nel campo della computer vision. Tramite la combinazione di filtri convoluzionali, operazioni di pooling e livelli totalmente connessi sono in grado di imitare al meglio l'organizzazione di una corteccia celebrale reale. Come quest'ultima infatti vengono estratte le caratteristiche importanti dall'immagine, semplificando il lavoro per l'ultima parte della rete che deve poi compiere il vero ragionamento.

Questo documento presenta un'analisi della rete EnzyNet, una rete che tramite l'uso di CNN tridimensionali classifica vari enzimi, spesso molto diversi tra loro, nelle 6 classi enzimatiche standard, utilizzando un'immagine a bassa risoluzione della loro "backbone" ovvero la struttura semplificata dei loro atomi e relative posizioni nello spazio.

# Indice

# Elenco delle figure

# Elenco delle tabelle

# Introduzione

All'interno della Protein Data Bank (PDB) sono conservati i dati di moltissimi enzimi. Ogni giorno questo numero aumenta e lo fa ad un ritmo sempre maggiore grazie anche alle nuove tecnologie che permettono di ottenere le strutture di queste molecole sempre più velocemente. Ultimamente è quindi sorta la necessità di avere una procedura sistematica, affidabile e rapida per assegnare ad ogni nuova aggiunta all'interno della PDB la propria classe di appartenenza, scegliendo tra le 6 classi enzimatiche standard.

Questi enzimi, anche detti catalizzatori biologici poiché svolgono la medesima funzione dei catalizzatori presenti in chimica, sono fondamentali per gli esseri viventi e, in particolare, per tutti i loro processi che coinvolgono reazioni con alta energia di attivazione. Le grandi quantità di dati in entrata necessitano dunque di strumenti adatti per essere processati: nasce così la bioinformatica. La bioinformatica si pone l'obiettivo di sviluppare modelli e metodi che ambiscono a comprendere le funzioni di tali catalizzatori, e più in generale ambiscono a comprendere e simulare le funzioni biologiche in toto in maniera molto più veloce ed efficiente di come si è si fatto finora.

L'assegnazione della funzione enzimatica alle proteine in un genoma è uno dei primi passi essenziali della ricostruzione metabolica, di fondamentale importanza per la biologia, la medicina, la produzione industriale e gli studi ambientali. Con la disponibilità sia di dati che di una potenza di calcolo entrambi sempre in crescita, gli approcci che vedono il deep learning come protagonista, come le reti neurali convoluzionali, si sono rivelati molto efficaci e hanno superato gli approcci

tradizionali.

L'obiettivo che ci si pone è quindi sviluppare (in questo caso analizzare ed aggiornare) un classificatore che per l'appunto assegni a ciascun enzima la propria classe di appartenenza.

# Capitolo 1

# Reti Neurali Convoluzionali

## 1.1 Definizione di CNN

Le reti neurali convoluzionali, in sigla CONVOLUTIONAL NEURAL NETWORK (CNN), sono un tipo di rete neurale feed-forward, ovvero sono reti in cui non è presente alcun ciclo formato da connessioni tra i neuroni, differentemente da quanto accade per le discendenti RNN. Esse imitano l'organizzazione della corteccia visiva presente negli animali, difatti i loro filtri si occupano ciascuno di estrarre delle feature particolari relative ad una data immagine per poi passarle ai layer finali fully connected che eseguono la vera classificazione. Già dal 1998 le CNN vengono applicate con buoni risultati per problemi semplici, come il riconoscimento di singoli caratteri o oggetti a bassa risoluzione. Con AlexNet si ebbe un cambiamento radicale, grazie anche all'enorme disponibilità di dati che caratterizza il nostro periodo. Si cominciò a pensare al concetto di BigData, per l'appunto dataset enormi, in genere etichettati, che rappresentano molto bene input e output di quello che deve poi essere la rete neurale.

Con l'avvento delle CNN si ebbe poi un ulteriore passaggio, uno dei più importanti in campo scientifico, che spesso è ciò che riesce a mantenere un progetto vivo: la commercializzazione.[6] Le CNN infatti sono risultate tanto ottimali da essere usate in ambiti commerciali dove l'affidabi-

lità è indispensabile, basti pensare ai moderni veicoli a guida autonoma.

Essendo una rete feed-forward, una CNN presenta un blocco di input, i layer nascosti che estraggono le feature e fanno un pre processing dei dati ed infine un blocco di output che esegue effettivamente la classificazione. Ai layer nascosti si aggiungono le funzioni di attivazione e i layer di dropout che svolgono una funzione fondamentale nel rendere queste reti il più simili possibile a quelle biologiche.

Gli strati di una CNN sono organizzati in una gerarchia: il primo livello, quello di input, è connesso ai singoli pixel dell'immagine in ingresso, mentre gli ultimi sono generalmente livelli completamente connessi che da soli potrebbero costituire una rete a sé, detta percettrone multistrato. Nei livelli intermedi vengono usati invece connessioni locali e pesi condivisi da più neuroni. Qui i neuroni sono connessi solo localmente a quelli del livello precedente, ciò permette di fargli processare esclusivamente una data porzione dell'immagine.

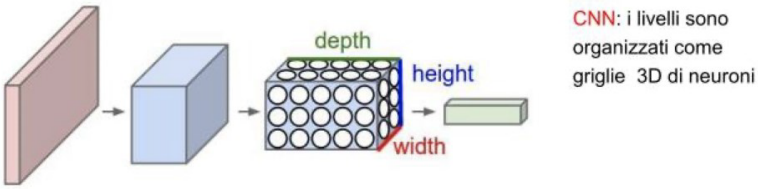



**Figura 1.1:** Organizzazione gerarchica di una CNN

Nelle reti neurali tradizionali si usa la moltiplicazione tra matrici, in cui una delle due matrici è l'input del livello e l'altra è una matrice di parametri, denominati anche pesi poiché vanno a determinare che peso impone un dato input su un dato output. Questo meccanismo implica un'interazione completa tra input di un livello e output del precedente. Per quanto riguarda le CNN invece le interazioni non riguardano tutti i neuroni insieme bensì avvengono delle interazione limitate, e questo può anche essere dedotto dalla dimensione del kernel che è solitamente molto più piccola dell'input. Questo fa sì che il numero di parametri da memorizzare sia di gran lunga

inferiore e si riduca per cui la necessità di risorse per processare una data rete e di conseguenza si migliorano le prestazioni.[2]

## 1.2 Operazioni di una rete neurale convoluzionale

La classificazione tramite CNN è composta solitamente dalla cascata di 3 operazioni:

- Convoluzione

  Si tratta di una delle più rilevanti operazioni in ambito del digital image processing. In essa possiamo osservare come vengano applicati dei filtri digitali, dei quali approfondiremo in seguito. Un filtro viene fatto scorrere sulla matrice di input e viene calcolato il prodotto matriciale, in quanto anch'esso è una matrice. Da qui si ricava un numero in uscita che rappresenterà l'output di quel filtro per quella data porzione di input. L'obiettivo di un filtro di questo tipo è comprendere gli schemi che sono contenuti in un'immagine. Per semplificare, essi possono essere cerchi, quadrati o più genericamente curve.

  Spesso questi filtri quando scorrono sulla matrice di input si spostano di più di un'unità per volta. Questo numero di unità è detto stride. [7] In tal modo si riduce la dimensione di output e parallelamente il numero di parametri, semplificando ulteriormente la rete.

  Il quesito che ci poniamo è dunque il seguente: come trattare i casi limite ovvero i pixel sul bordo?

  Ci sono attualmente due strade che vengono scelte solitamente: non considerarli oppure considerare la parte di filtro che agisce fuori dalla matrice come agente su pixel fasulli sempre impostati a 0. Quest'ultima operazione è detta padding della matrice. Le dimensioni dell'output possono essere definite dalla seguente formula:

$$W_{out} = \frac{W_{in} - F + 2 * Padding}{Stride} + 1$$

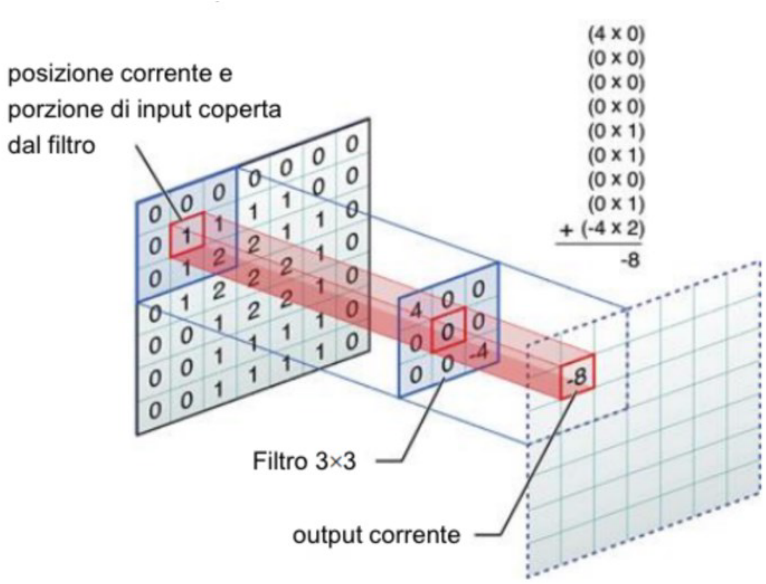
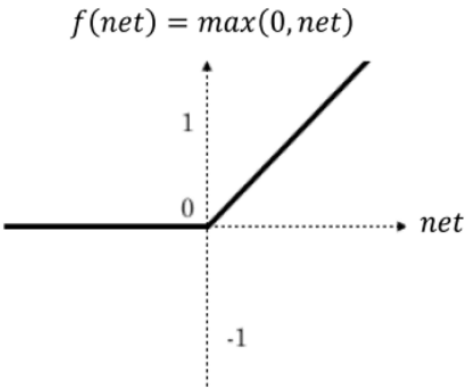



**Figura 1.2:** Operazione di convoluzione

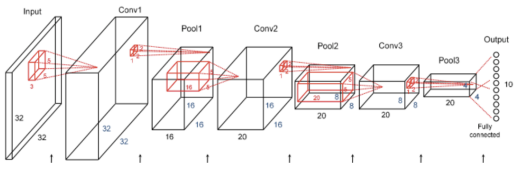
- Detector Stage

  Qui i neuroni trasmettono il loro output lineare ad una funzione molto spesso non lineare. La funzione attualmente più utilizzata nelle reti convoluzionali e molto usata anche negli altri tipi di reti è la ReLu, acronimo di Rectified Linear Unit. Questa funzione, descritta come $max(0, in)$, restituisce 0 se riceve un input inferiore a 0, l'input stesso altrimenti. Ultimamente si sta spesso optando per una derivata della rectified linear (ReLu), la LeakyReLu che opta a non azzerare completamente input negativi anche se vengono considerati con un fattore moltiplicativo sempre inferiore ad 1 e pertanto sempre meno della corrispettiva parte positiva. Data la natura della ReLu, questa funzione permette di attivare un numero inferiore di neuroni che semplifica la rete.

- Pooling




**(a)** Il grafico della funzione ReLu




**(b)** Esempio di una CNN completa

**Figura 1.3**

Il pooling è un'operazione che aggrega più informazioni da un dato input per generare output di dimensioni inferiori. Questa tecnica aiuta la rete a non discriminare tra input molto simili ma leggermente traslati: [5] un caso molto comune quando si parla di CNN. Aggregando le varie informazioni si va ad agire sulle singole feature map, lasciando il loro numero invariato tra input ed output. Come operatori di aggregazione, anche qui come nel padding visto precedentemente, sono 2 i principali: average pooling e max pooling. Come suggeriscono i due nomi, il primo calcola la media aritmetica dei valori in input e la ritorna come output, il secondo invece prende il massimo di questi valori e lo ritorna anch'esso come output. Il secondo approccio è spesso preferito all'altro, tuttavia è quello che porta ad una maggiore perdita di informazione in quanto esclude in toto tutti i valori tranne uno. Il primo, nonostante mantenga più informazione è quello che poi porta spesso la rete a considerare dettagli irrilevanti come rilevanti, anche solo in parte, il che riduce l'efficienza della rete stessa. [4]

# Capitolo 2

# Enzimi

## 2.1 Introduzione agli enzimi

Gli enzimi rientrano nella categoria delle macromolecole, nello specifico la maggior parte di essi rientra nelle proteine globulari, mentre una piccola parte è costituita da particolare catene di RNA dette ribosomi. Questi enzimi hanno la funzione di catalizzatore biologico in quanto all'interno degli esseri viventi svolgono proprio un'attività catalitica. È importante sottolineare che queste molecole non alterano in alcun modo gli equilibri della reazione che catalizzano ma inducono un incremento nella rapidità della reazione in quanto aiutano ad abbassare l'energia di attivazione. Tale energia è talvolta così elevata che se non agissero gli enzimi, molte delle reazioni che governano gli esseri viventi non potrebbero avvenire, con un chiaro impatto negativo sulla vita stessa. Questo enzimi, in particolare nel caso di enzimi globulari idrosolubili, hanno bisogno per svolgere la loro funzione di ulteriori ausili chiamati cofattori che possono essere semplici ioni metallici oppure anche molecole organiche più complesse.

Il processo di catalisi avviene tramite il legame dei substrati dei reagenti ad una specifica regione dell'enzima che per questo viene chiamato sito attivo. È questo il sito che poi determina la categorizzazione tramite l'ECN (Enzyme Commission Number). Ogni enzima è caratterizzato da un

| Numero | Classe | Tipo di reazione catalizzata |
|---|---|---|
| EC1 | Ossidoreduttasi | Trasferimento di elettroni (ioni ioduro $H^-$ o atomi di H) |
| EC2 | Transferasi | Reazioni di trasferimento di gruppi funzionali |
| EC3 | Idrolasi | Reazioni di idrolisi (trasferimento di gruppi funzionali all'acqua) |
| EC4 | Liasi | Addizione di gruppi a legami doppi o formazione di legami doppi mediante eliminazione di gruppi |
| EC5 | Isomerasi | Trasferimento di gruppi all'interno di molecole per formare isomeri |
| EC6 | Ligasi | Formazione di legami $C-C$, $C-S$, $C-O$, e $C-N$ mediante reazioni di condensazione accoppiate alla scissione di ATP |

**Tabella 2.1:** Corrispondenza $ECN-Funzione$

codice gerarchico che è costituito da ECXXXX dove XXXX sono 4 numeri, ciascuno indicante una caratteristica dell'enzima. Il primo dei 4 numeri indica sempre il tipo di reazione che viene catalizzata, gli altri 3 sono specifici e caratterizzano ciascun enzima.

## 2.2 Caratteristiche degli enzimi

Gli enzimi non sono gli unici catalizzatori esistenti, tuttavia si differenziano dai catalizzatori non enzimatici per principalmente tre caratteristiche:

- Efficienza

  Gli enzimi possiedono un'efficienza catalitica maggiore di vari ordini di grandezza rispetto agli altri. Questo porta le reazioni a svolgersi in tempi brevissimi e soprattutto fa sì che nessuna interazione enzima-substrato fallisca. Ciò si traduce con una reazione andata a buon fine per ogni interazione che si viene a creare, riducendo ancor di più il rischio di fallimento nel caso in cui vi sia la necessità di più interazioni per ogni singola reazione.

- Specificità

  Gli enzimi sono generalmente molto selettivi sul tipo di reazione che catalizzano: solitamente ogni enzima catalizza un solo specifico tipo di reazione. Questo fa sì che vi sia la necessità di molti enzimi, almeno in numero pari alle reazioni da catalizzare. Tuttavia una maggiore specificità porta l'enzima a svolgere la propria funzione alla perfezione.

- Regolabilità

  Gli enzimi hanno la possibilità di variare il proprio stato di attività anche in base alla concentrazione di substrato e/o di prodotti. Questo genera un meccanismo di autoregolazione e di feedback che porta gli enzimi a non attivarsi mai in caso di non necessità. Ciò riduce lo spreco di risorse ed evita possibili complicazioni dovute alla maggiore o minore concentrazione di substrati/prodotti.

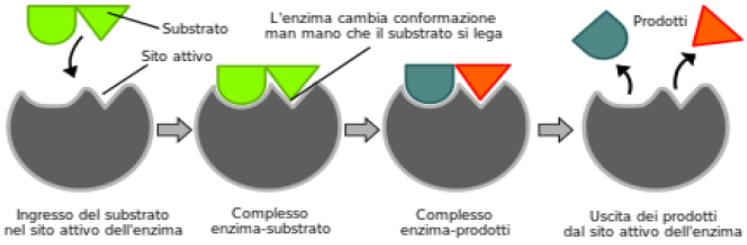



**Figura 2.1:** Complesso enzima-substrato

Il meccanismo tramite cui gli enzimi lavorano per abbassare l'energia di attivazione delle reazioni consiste nel formare un complesso enzima-substrato tramite il legame con i reagenti, spostandoli e orientandoli per portarli nella posizione migliore per far avvenire la reazione. L'adattamento del substrato con l'enzima controlla poi la selettività per il substrato stesso e la resa del prodotto.[10] L'attività degli enzimi è spesso regolata da fattori quali:

- Aumento della concentrazione del substrato

  Una maggiore concentrazione di substrato porterà, a parità di concentrazione di enzima, una velocità maggiore di reazione. Questo porta un minore consumo di energia e risorse quando non necessario.

- Inibizione enzimatica irreversibile

  A volte può capitare che si formino legami covalenti, molto difficili da rompere, tra l'enzima e una qualche altra molecola che può essere di varia natura: uno scarto, un prodotto di un

agente esterno, un ormone oppure addirittura può essere una molecola prodotta dallo stesso enzima con il preciso scopo di inibire l'enzima, in questo caso si parla di inibizione suicida.

- Inibizione enzimatica reversibile

  Questo tipo di inibizione è dovuta a molecole che si legano all'enzima in maniere non sufficientemente forte da renderlo inutilizzabile per sempre. In caso di inibitori competitivi, si hanno più molecole che si legano allo stesso enzima, dove almeno una di queste è una molecola dedicata a ridurre l'attività enzimatica, che accade spesso ad esempio con gli ormoni. Nel caso di inibizione non competitiva invece si avrà una sola molecola, solitamente il prodotto, che si lega con l'enzima in modo da rallentare la velocità della reazione [**a3**] e quindi portando la concentrazione di prodotto ad abbassarsi.

## 2.3 Struttura degli enzimi

Negli enzimi proteici, la struttura che li compone può essere suddivisa in 4 differenti livelli di organizzazione:

- Struttura primaria ovvero la sequenza di aminoacidi
- Struttura secondaria ovvero la conformazione spaziale delle catene di aminoacidi, spesso conosciute con il nome di alfa e beta eliche
- Struttura terziaria ovvero la conformazione spaziale di una intera catena polipeptidica
- Struttura quaternaria ovvero la struttura assunta da più catena polipeptidiche legate tramite legami deboli tra loro [1]

Due proteine distinte potrebbero mostrare macrostrutture molto simili. Questo potrebbe essere dovuto due principali motivi: avere un antenato comune che poi l'evoluzione ha fatto evolvere in due rami distinti oppure presentano due enzimi che si sono evoluti per catalizzare la stessa reazione o avere lo stesso sito attivo. In questo caso si parla di evoluzione convergente, contrapposta a quella

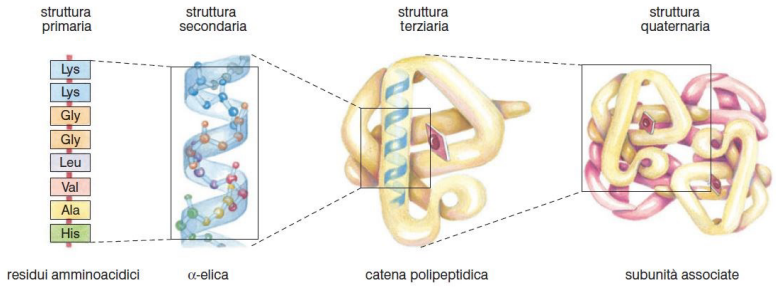



**Figura 2.2:** Suddivisione della struttura di una proteina

precedente detta divergente.

All'interno di una stessa famiglia proteica, la struttura tridimensionale della molecola viene conservata in maniera migliore rispetto alla sequenza di aminoacidi che compongono le proteine. La sequenza di aminoacidi infatti può, per cause evolutive, doversi scontrare con degli eventi di mutazione casuale e puntuale, elisione o inserzioni. Questo fa sì che sia sufficiente solamente il 20% degli aminoacidi a determinare il grado di parentela, e quindi la famiglia di appartenenza, di due proteine distinte.

# Capitolo 3

# EnzyNet

## 3.1 Cos'è EnzyNet

EnzyNet è una Convolutional Neural Network 3D di nuova generazione. Questa rete si pone l'obiettivo di creare un classificatore che predica in modo molto accurato l'ECN di un dato enzima basandosi esclusivamente sulla sua struttura tridimensionale, senza nemmeno tener conto di quali atomi siano presenti nella struttura. EnzyNet infatti necessita esclusivamente della spina dorsale della molecola ovvero la posizione dei principali atomi nello spazio, senza tener conto di gruppi accessori, necessari alla molecola dal punto di vista chimico ma che non determinano nulla per quanto riguarda la funzione della stessa. Questa rete si basa su un dataset di 63558 elementi, tutti provenienti dalla PDB.

Una cosa degna di nota è sicuramente la disponibilità del codice di EnzyNet e del suo dataset. Difatti, troppo spesso si tende a rendere disponibili i risultati senza dare il sorgente, interropendo così la catena del metodo scientifico che definisce esperimento come una procedura ripetibile da chiunque. Rendere il codice closed source di sicuro non rende un esperimento non valido ma nella pratica lo rende non ripetibile.

## 3.2 Preprocessamento dei dati

Il team di EnzyNet ha stabilito che la struttura degli enzimi non è esclusivamente quella osservabile in natura quindi ha deciso di apportare una modifica sostanziale alla rappresentazione degli stessi. Nello specifico è stato deciso di perseguire una modalità di rappresentazione che si basa sulla struttura spaziale tridimensionale degli enzimi all'interno di volumi cubici di lato prefissato. Tale metodo è in grado, se utilizzato a dovere e con il giusto algoritmo di pre-processing, di aumentare notevolmente le prestazioni della rete, non solo in termini di velocità in quanto è necessaria una rete meno complessa, ma anche in termini di accuratezza. Difatti il team di EnzyNet è riuscito a raggiungere l'80% di accuratezza nel dataset di test.

La creazione di queste rappresentazioni, basate su volumi cubici, necessita di alcune operazioni preliminari. La prima cosa fondamentale da stabilire è una risoluzione standard che sia uguale per tutte le rappresentazioni di enzimi appartenenti al dataset. Questo è necessario perché gli enzimi possono avere dimensioni molto differenti tra loro e una conversione impropria porterebbe molto probabilmente ad una perdita consistente di informazioni in specifiche situazioni. Dalla backbone inoltre è stato deciso di rimuovere in toto le catene laterali, che si è visto non portano ulteriore informazione per quanto concerne la funzione dell'enzima. Ciò semplifica le rappresentazioni, eliminando dettagli inutili che potrebbero portare la rete a carpire pattern sbagliati o superflui. Eliminare informazione inutile è un ottimo metodo per il risparmio di risorse e i tempi di elaborazione. Vengono quindi presi solamente gli atomi principali quali carbonio, azoto e calcio. È necessario tener conto che una proteina, seppur ruotata, svolge sempre la medesima funzione. Tuttavia addestrare una rete non tenendo conto di ciò potrebbe portare la rete a fare assunzioni circa l'orientamento delle molecole, che porterebbe ad errori. Per questi motivi è stata introdotta una certa casualità circa le rotazioni lungo i 3 assi principale in modo da evitare spiacevoli inconvenienti.

Abbiamo visto che gli enzimi possono avere dimensioni molto varie, alcune volte un enzima può essere anche 10 volte più grande rispetto ad uno più piccolo. Per questo è necessario adattare la loro struttura alla struttura del volume in cui devono essere rappresentati. Si possono percorre due strade in questo caso: riscalare ogni atomo singolarmente affinchè rientri nel volume oppure scalare tutti gli atomi di un valore predefinito indipendentemente dalla dimensione dei singoli. Il primo approccio, seppur sia quello che trattiene il maggior numero di informazioni per quanto concerne la struttura dell'enzima stesso, non tiene conto di una cosa fondamentale: la dimensione spaziale. Se scalassimo tutti gli enzimi indipendentemente avremmo delle molecole perfettamente rappresentate dal punto di vista dei loro atomi ma la rete non avrebbe modo di classificarli basandosi sulla loro dimensione, un elemento molto importante per capire la loro funzione, non ci sarebbe quindi modo di fare confronti tra un enzima e l'altro e questo porterebbe certamente a classificazioni errate. Il secondo approccio invece è da preferirsi in quanto la dimensione viene preservata. Nonostante ciò, neanche il secondo approccio risulta perfetto in quanto taglia di netto la parte di struttura che fuoriesce dal volume e quindi perde informazione talvolta importante. Ora è necessario trovare un fattore di scala ossia trovare quel fattore che mantenga la maggiore quantità di informazioni in assoluto attraverso tutti gli enzimi rappresentati. [3] Si è scelto di stabilire quindi un $R_{max}$ che fosse contenuto all'interno del volume. Infine, per evitare ulteriormente la perdita di informazioni si è deciso di rappresentare gli enzimi traslandoli, facendo combaciare il loro baricentro con quello del volume cubico in modo da contenere il maggior numero possibile di atomi.

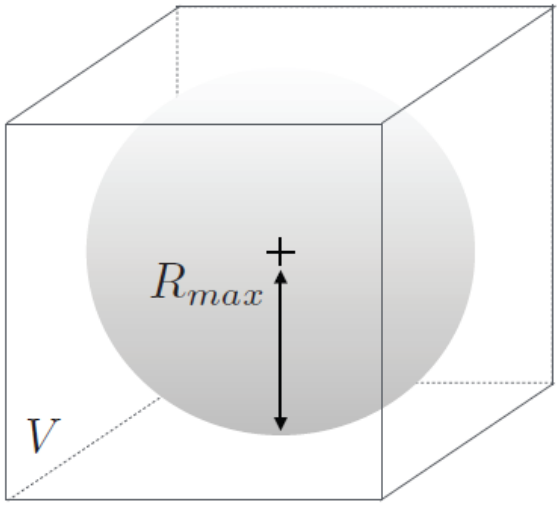



**Figura 3.1:** Visualizzazione del raggio $R_{max}$ rispetto al cubo V

## 3.3 Modellazione 3D

Come già analizzato, tutti gli enzimi necessitano di essere scalati allo stesso modo e con una precisione sufficiente. Per raggiungere tale obiettivo il team di EnzyNet è ricorso ad una trasformazione omotetica con centro in S, che viene definito come il centro del cubo V e con raggio $\lambda$:

$$\lambda = \lfloor \frac{l}{2} - 1 \rfloor * \frac{1}{R_{max}}$$

Con questa tecnica di rappresentazione è sorto un piccolo problema: nel caso il cui il lato l sia un valore troppo basso e la griglia non sia sufficientemente dettagliata, si rende necessario aggiungere del dettaglio artificialmente apportando un'interpolazione degli atomi principale connessi tra loro e quindi consecutivi. Questa operazione avviene mediante l'aggiunta di un numero di nuovi punti p calcolato secondo la seguente relazione:

$$\frac{(p-k+1)*\vec{A_t} + k*\vec{A_{l+1}}}{p+1}$$

dove k deve essere compreso tra 1 e p. Nella seguente figura si nota molto chiaramente come varia la forma e la struttura di un dato enzima a seconda della lunghezza del lato l del cubo. [9]

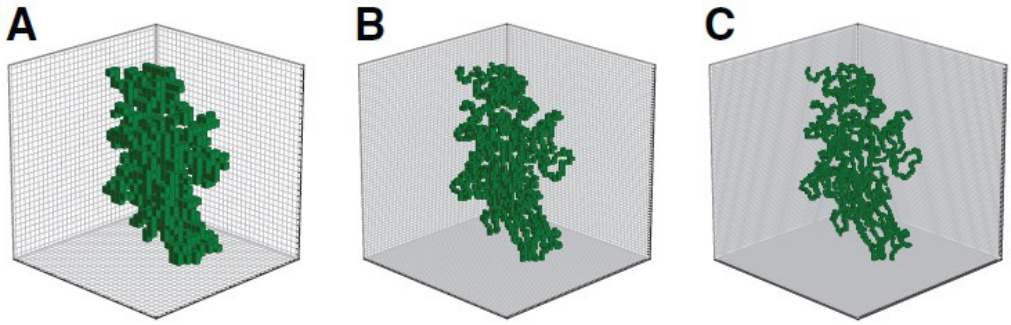



**Figura 3.2:** Variazione della risoluzione per L = 32 (A), 64 (B), 96 (C)

Per rendere uniformi tutte le proteine contenute all'interno del dataset è necessario che vengano applicate delle trasformazioni, al massimo 7, ovvero $2^3 - 1$, che preservino la componente principale della molecola lungo gli assi e che rendano i volumi tra loro comparabili. [8]

Nella seguente figura è possibile avere un'idea, sotto forma di pseudo codice, dell'algoritmo che preprocessa gli enzimi nel dataset prima che essi vengano visti dalla rete:

**Dati:** N enzimi, una griglia di lato l, un fattore di trasformazione λ, un numero di interpolazioni p e una probabilità di rotazione $p_{flip}$ rispetto ad ogni asse
**Input:** coordinate contenute nei files PDB
**Output**: volumi dei voxels binari rappresentanti l'occupazione degli atomi

**foreach** enzima N del training set **do**
  Step 1: estrazione delle informazioni strutturali
  Estrarre le coordinate degli atomi principali dal loro file PDB
  Step 2: interpolazione
  Interpolare gli atomi consecutivi mediante p nuovi punti
  Step 3: modifica delle dimensioni
  Centrare il baricentro S delle coordinate nel punto (0,0,0)
  Operare la trasformazione omotetica di ogni punto con centro S e fattore λ
  Step 4: orientamento dell'enzima
  Trasformare l'analisi della componente principale (PCA)
  Step 5: augmentation casuale
  **if** True con probabilità $p_{flip}$ **then**
    Ruotare le coordinate attorno all'origine lungo l'asse – x
  **if** True con probabilità $p_{flip}$ **then**
    Ruotare le coordinate attorno all'origine lungo l'asse - y
  **if** True con probabilità $p_{flip}$ **then**
    Ruotare le coordinate attorno all'origine lungo l'asse - z
  Step 6: voxelizzazione
  Centrare il baricentro S delle coordinate nel punto $\left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}\right)$
  Trasformare i punti delle coordinate in voxels binari

**Figura 3.3:** pseudo codice del preprocessamento dati

# Capitolo 4

# Addestramento della rete neurale

Al fine di sviluppare farmaci e agenti molecolari, risulta fondamentale conoscere la classe EC di uno specifico enzima. La rete EnzyNet è un tipo di rete neurale convoluzionale tridimensionale usata proprio per ricavare l'ECN tramite la classificazione della struttura tridimensionale di un enzima. Per eseguire correttamente la classificazione viene convertita la rappresentazione precedentemente calcolata in una rappresentazione tridimensionale binaria della struttura della spina dorsale dell'enzima e successivamente la rete viene addestrata con una parte degli enzimi del dataset, l'80% nello specifico, in quanto la restante parte è necessaria per validare la rete, ossia capire se la rete sta intuendo davvero gli schemi o sta solo memorizzando quelli che vede.

Durante l'addestramento di EnzyNet si sono testate varie strutture per la rete. Questo era necessario per capire quale più si adattasse a tale compito in quanto le tecniche attuali non ci permettono di definire con precisione quali reti siano più corrette per un dato compito. Durante i test, come in molte altri occasioni usando le reti neurali, si può intercorrere in un paio di problemi: il primo consiste nell'underfitting ovvero nell'ottenere prestazioni scarse durante l'addestramento, come spesso succede a causa di una struttura della rete troppo semplice. Il secondo, al contrario, consiste nell'overfitting, ovvero nell'ottenere prestazioni molte buone ma solo nel training set, mentre nel test di validazione si ottengono risultati scarsi. Questo è spesso dovuto ad una rete troppo

complessa che è in grado di imparare e memorizzare nei propri pesi le strutture degli enzimi con cui viene addestrata e i relativi ECN.

La rete EnzyNet è una rete formata nel modo seguente:

- Il layer di input

  Un layer cubico di lato 32 (o maggiore ma per la rete finale si è scelto 32) che prende in input la struttura tridimensionale dell'enzima.

- Un layer convoluzionale

  Un layer composto da 32 filtri di lato 9, che generano 32 matrici tridimensionali di lato 12, applicando i filtri di questo strato con stride 2.

- Un dropout 0.2

- Un secondo layer convoluzionale

  Un layer composto da 64 filtri di lato 5, che generano 64 matrici tridimensionali di lato 8, applicando i filtri di questo strato con stride 1.

- Un dropout 0.3

- Un layer di maxpooling

  Un layer maxpooling che agisce tramite una matrice cubica di lato 2.

- Un layer fully connected

  Un layer monodimensionale compoisto da 128 neuroni

- Un dropout 0.4

- Un secondo layer fully connected di output Un layer con un numero di neuroni pari alle classi esistenti ovvero 6. Questo è il layer finale che effettua la vera classificazione.

- Un layer softmax

  Questo layer trasforma l'output del precedente livello in probabilità la cui somma è pari ad 1.

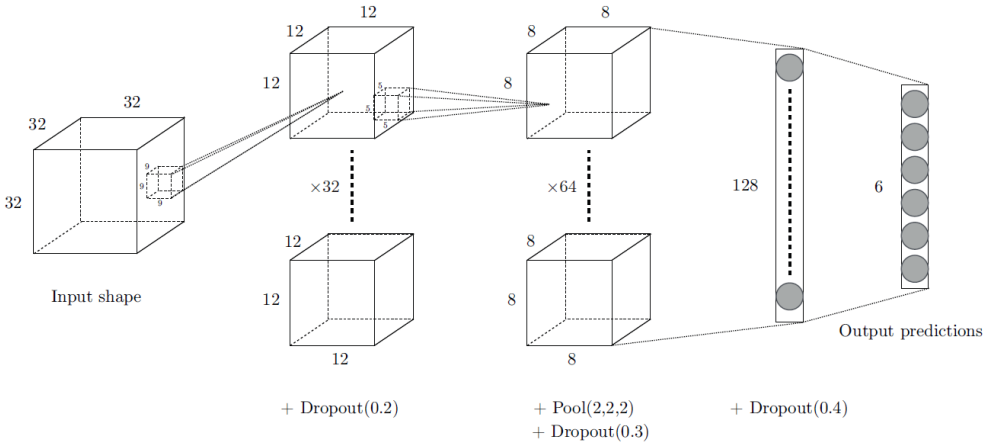



**Figura 4.1:** Architettura della rete EnzyNet

## 4.1 Problemi insorti durante l'addestramento

Durante l'addestramento sono insorti vari problemi, principalmente dovuti all'età del codice e al cattivo mantenimento dello stesso.

Il codice seppur funzionante al momento della sua pubblicazione, risultava non più tale nel momento della scrittura di questo documento. Il problema di base erano le dipendenze su cui il codice si basava, alcune non erano specificate e nessuna aveva segnata la relativa versione. Ciò ha comportato un maggiore impegno per ricercare e ritrovare le versioni originali, cercando di collegare data di pubblicazione del paper con la data di pubblicazione delle varie versioni delle librerie.

Un altro problema è stato sicuramente la cattiva qualità di varie parti del codice che utilizzavano funzioni cosiddette deprecate, ovvero funzioni che non dovrebbero essere più usate poiché obsolete. Questo, seppur non di fondamentale importanza, ha portato la ricerca delle versioni originali delle librerie fuori strada, poiché si supponeva che il codice originale non usasse funzioni deprecate al momento della scrittura.

Un ultimo problema è stato l'enorme frammentazione del codice originale e la sua cattiva documentazione che hanno reso difficile capire come generare il dataset, come addestrare la rete e come eseguire una classificazione.

Per tutti questi problemi si è in primo luogo optato per riscrivere interamente il codice, utilizzando un altro linguaggio che semplificasse lo sviluppo e che rendesse facile la mantenibilità. Purtroppo questo approccio, al momento della scrittura di questo documento, sembra non funzionare in quanto, dopo essere riusciti a far eseguire il codice originale, si è notata una certa discrepanza enorme nelle prestazioni a parità di rete, questo ha portato a concentrare gli sforzi sul codice originale.

# Capitolo 5

# Risultati ottenuti

Di seguito sono espressi i risultati ottenuti con EnzyNet, sia nella versione rifatta in Matlab, sia nella versione originale in python.

## 5.1 Versione Matlab

Con la versione matlab si era inizialmente partiti ottenendo non più del 40% in validazione. Questo era di certo un risultato che si allontanava di molto da quello originale. Si è pensato allora di espandere il set da 6000 unità circa alle 60000 di tutto il dataset originale. Questo ha portato un incremento delle prestazioni a quasi il 50%. Le prestazioni erano migliorate ma non in maniera ottimale da renderlo valido. Ci si è poi accorti di piccoli dettagli nella rete che contrastavano con la rete originale, dettagli di poco conto che hanno contribuito ad aumentare l'accuratezza di poco, circa il 2%. Infine, si è deciso di applicare più varietà nel dataset e, in particolare, alla parte di data augmentation, dove gli enzimi venivano ruotati, scalati e posizionati. Traducendo ciò in un aumneto del numero di queste trasformazioni per ogni enzima si è arrivati ad un'accuratezza del 65-70%, molto meglio del 40% iniziale ma sempre non ottimale. Dopo questo risultato ci si è decisi di spostare gli sforzi nel far eseguire il codice originale.

## 5.2 Versione Python (originale)

La versione originale ha ottenuto, dopo i numerosi sforzi per adattarla alla modernità, buoni risultati, circa l'80% con un minimo del 75% che è decisamente più auspicabile rispetto ai risultati precedenti. Con questo codice ci si è spinti oltre provando varie forme di data augmentation, in particolare si sono ottenuti risultati anche fino all'83% applicando varie trasformazioni.

# Conclusioni

Col trascorrere del tempo, la potenza computazionale è in crescita esponenziale. La quantità di dati a disposizione, le nuove tecnologie e le nuove tecniche di progettazione e addestramento, fanno sì che l'approccio di deep learning sia maggiormente considerato nei problemi di classificazione. È stato dimostrato che enzimi con funzione simile tra loro presentano molte volte anche strutture simili tra loro. L'assegnazione della funzione enzimatica alle proteine in un genoma, è uno dei primi passi essenziali della ricostruzione metabolica, fondamentale per la biologia, la medicina, la produzione industriale e gli studi ambientali.

Nel presente studio è stata selezionata una griglia relativamente piccola, 32x32x32 che tende a perdere dettagli che possono essere importanti per ottenere quel 20% che manca ad arrivare alla perfezione. Purtroppo non si è stati in grado di crescere con la dimensione della griglia per costrizioni dovute alle scarse risorse computazionali.

Questo non è un lavoro terminato ma viene lasciato ai posteri, sperando che qualcuno lo prenda in mano e riesca a migliorarlo, magari anche grazie a calcolatori più potenti che in futuro potrebbero essere alla portata di tutti. Questo progetto ha di certo del potenziale e potrebbe portare una svolta in vari ambiti, in particolare nella medicina grazie alla velocità con cui è in grado di elaborare gli enzimi.

# Glossario

**rectified linear (ReLu)** È una funzione di attivazione nata per risolvere alcuni problemi delle funzioni di attivazione usate nelle reti MLP, come la sigmoide, in quanto essa ha problemi nella retropropagazione del gradiente. La funzione è definita come

$$f(x) = \max(0, x)$$

La funzione restituisce 0 se il valore è minore di zero altrimenti restituisce il numero stesso. 6

**convolutional neural network (CNN)** Una rete neurale convoluzionale (ConvNet/CNN) è un algoritmo di Deep Learning che prende in input un'immagine, ai vari input vengono assegnati i pesi e un bias in modo che la rete sia in grado di cogliere gli aspetti e gli oggetti dell'immagine ed essere in grado di differenziare gli uni dagli altri. Come si deduce dal nome le CNN fanno ampio uso della convoluzione

La pre elaborazione richiesta in una CNN è molto più bassa rispetto ad altri algoritmi di classificazione. Mentre nei metodi primitivi i filtri sono costruiti a mano, con abbastanza addestramento, le CNN hanno la capacità di imparare questi filtri.

L'architettura di una CNN è molto simile a quella del modello di connettività dei neuroni nel cervello umano ed è stata ispirata dall'organizzazione della corteccia visiva. I singoli

neuroni rispondono agli stimoli solo in una regione ristretta del campo visivo noto come campo recettivo. Un insieme di tali campi si sovrappone per coprire l'intera area visiva.

Le reti CNN hanno diverse applicazioni nel riconoscimento di immagini, video e audio, nei sistemi di raccomandazione, nell'elaborazione del linguaggio naturale e, recentemente, in bioinformatica. 3

**gradient descent (GD)** È il principale algoritmo di error backpropagation, la minimizzazione dell'errore avviene attraverso passi in direzione opposta al gradiente.

30

**stochastic gradient descent (SGD)** È un metodo iterativo per l'ottimizzazione di funzioni differenziabili, è l'approssimazione stocastica del metodo di gradient descent (GD) quando la funzione costo è una somma. È ampiamente usato nell'allenamento dell'intelligenza artificiale 31

**convoluzione** È un operatore matematico che a partire da funzioni $f(t)$ e $g(t)$, ne restituisce una terza che rappresenta come la forma di una delle due funzioni una influisce sull'altra. 29

**fully connected** I livelli completamente connessi in una rete neurale sono quei livelli in cui tutti gli input di un livello sono collegati a ogni unità di attivazione del livello successivo. Normalmente nei modelli di apprendimento automatico più diffusi, gli ultimi livelli sono livelli completamente connessi che compilano i dati estratti dai livelli precedenti per formare l'output finale. È il secondo livello più dispendioso in termini di tempo, dopo il Convolution Layer. 22, 31

**overfitting** Succede quando il sistema tende a identificare relazioni nell'insieme di addestramento che non valgono in generale. nel senso che il sistema impara a memoria i pattern, questo porta a ottimi risultati nel data set e pessimi risultati nel test set o negli usi reali. 21

**pesi** Detti anche sinapsi è il valore di quanto è importante una determinata feature in ingresso al neurone, i pesi vengono modificati da un algoritmo di back propagation come il stochastic gradient descent (SGD). 4

**softmax** Alla fine della rete neurale profonda c'è un livello finale softmax, che effettua una vera e propria classificazione: consiste in $n$ neuroni, uno per ogni classe, ciascuno connesso a tutti i neuroni del livello precedente (fully connected). Essenzialmente è simile ai neuroni della MLP.

Il livello di attivazione $\text{net}_k$ dei singoli neuroni si calcola nel modo consueto, ma come funzione di attivazione per il neurone $k$-esimo (invece di Sigmoide o ReLu) si utilizza:

$$z_k = f(\text{net}_k) = \frac{e^{\text{net}_k}}{\sum_{c=1\ldots s} e^{\text{net}_c}}$$

dove i valori $z_k$ prodotti possono essere interpretati come probabilità: appartengono a $[0...1]$ e la loro somma è 1. Il livello SoftMax, quindi traduce i valori di net prodotti dall'ultimo livello della rete in probabilità.

È importante sottolineare che funzioni come Tanh, Sigmnoid, ReLu non forniscono probabilità in quanto:

$$\sum_{c=1\ldots s} z_c \neq 1$$

23

# Bibliografia


[1] Poleksic A. “Algorithms for optimal protein structure alignment.” In: *Bioinformatics 25*. 2009.

[2] Provino A. “Machine Learning  Data Science Blog. Convolutional Neural Network (CNN).” In: 2020. URL: https://andreaprovino.it/convolutional-neuralnetwork/.

[3] Amidi et al. “EnzyNet: enzyme classification using 3D convolutional neural networks on spatial representation.” In: (2018).

[4] Casadei C. “Reti convoluzionali”. In: Maggioli Developers, 2019. URL: https://www.developersmaggioli.it/blog/reti-convoluzionali/.

[5] Soriano D. “Come funziona una rete neurale CNN (Convolutional Neural Network).” In: 2019. URL: https://www.domsoria.com/2019/10/come-funziona-una-rete-neurale-cnnconvolutional-neural-network/.

[6] Lorenzo Govoni. “Semplice architettura di rete neurale convoluzionale”. In: URL: https://www.lorenzogovoni.com/architettura-di-rete-neurale-convoluzionale/.

[7] Geoffrey E. Hinton Krizhevsky A. Sutskever I. “ImageNet Classification with Deep Convolutional Neural Networks.” In: *Advances in neural information processing systems*. 2012.

[8] T. Kawabata. “MATRAS: a program for protein 3D structure comparison”. In: *Nucleic Acids Research 31*. 2003.

[9] John Q Trojanowski Johannes Brettschneider Virginia M.-Y Lee. “Neurodegene- rative disease concomitant proteinopathies are prevalent, age-related and APOE4- associated.” In: *Brain 141*. 2018.

[10] Sonderby SK Sonderby CK Nielsen H Winther O. “Convolutional LSTM Networks for Subcellular Localization of Proteins. Cham: Springer International Publishing”. In: 2015.