# Utilizzo di CNN e GNN per la Predizione dei Docking Score a Partire da Strutture Molecolari

Giorgia Roselli[1] - Kevin Attarantato[1], Sara Montagna[2]

**Sommario**
Questo progetto, realizzato per l'esame di *Deep Learning* presso *l'Università degli Studi di Urbino Carlo Bo*, si concentra sulla predizione dei docking score a partire da strutture molecolari rappresentate in formato SMILES (Simplified Molecular Input Line Entry System). Gli SMILES sono una notazione testuale per descrivere la struttura chimica delle molecole, utilizzata in questo progetto per rappresentare i dati.
L'obiettivo principale dello studio è sviluppare un modello predittivo per stimare il docking score delle molecole, un valore critico che indica l'affinità di legame tra una molecola e il suo target biologico.
Per raggiungere questo obiettivo, il progetto utilizza la libreria RDKit per trasformare le rappresentazioni SMILES in due formati computazionali principali: 'Molecular Fingerprint' sequenza binaria che codifica le proprietà molecolari e 'Molecular Graph' rappresentazione grafica che descrive gli atomi e i legami della molecola.
I dati così elaborati vengono poi utilizzati per addestrare due tipologie di reti neurali avanzate: 'Convolutional Neural Networks' (CNN) e 'Graph Neural Networks' (GNN).
Il progetto si propone di esplorare l'efficacia combinata di queste tecniche di deep learning per migliorare la capacità predittiva del modello e fornire stime accurate dei docking score.

**Keywords**
CNN – GNN – NeuralNetwork – Smiles – Docking Score – Deep Learning

[1] *Laurea Magistrale in Informatica Applicata, Università degli Studi di Urbino Carlo Bo, Urbino, Italia*
[2] *Docente di Applicazioni dell'Intelligenza Artificiale, Università degli Studi di Urbino Carlo Bo, Urbino, Italia*
*__Corresponding author__: g.roselli1@campus.uniurb.it - k.attarantato@campus.uniurb.it

## Introduzione

Nel panorama contemporaneo della ricerca farmaceutica, lo sviluppo di nuovi farmaci rappresenta uno dei processi più complessi, lunghi e costosi. Le metodologie tradizionali prevedono anni di sperimentazioni in laboratorio e ingenti investimenti economici, con un elevato tasso di fallimento, specialmente nelle fasi avanzate di sviluppo clinico. Le tempistiche spesso estese e i costi associati sono legati all'identificazione di molecole promettenti e alla verifica della loro efficacia e sicurezza. Questo approccio convenzionale, sebbene consolidato, è diventato sempre più inadeguato di fronte alla crescente complessità delle malattie moderne e alla necessità di sviluppare farmaci personalizzati.

L'integrazione delle tecnologie di intelligenza artificiale (IA), e in particolare del *deep learning*, sta trasformando radicalmente questo settore. Queste tecnologie offrono strumenti avanzati per analizzare grandi moli di dati chimici e biologici, identificando correlazioni e pattern che sarebbero difficilmente rilevabili con approcci tradizionali. Grazie a tali metodologie, diventa possibile accelerare i tempi di scoperta e ottimizzare le risorse economiche e computazionali, riducendo i tassi di fallimento e migliorando la precisione nella selezione dei candidati farmaci.

### 0.1 Drug Discovery e Virtual Screening

Il processo di *drug discovery* consiste nell'identificazione di nuove molecole che possano agire come farmaci, interagendo efficacemente con target biologici specifici, come proteine o enzimi coinvolti in patologie. Questo processo è caratterizzato da numerose fasi, dalla selezione iniziale di molecole candidate al loro perfezionamento e ottimizzazione. Una fase cruciale è il virtual screening, che permette di analizzare grandi librerie di composti chimici in modo computazionale, riducendo la necessità di test sperimentali costosi e lunghi.

Tra i metodi di *virtual screening*, il molecular docking riveste un ruolo fondamentale. Questo approccio simula l'interazione tra una molecola candidata ("ligando") e il target biologico, fornendo una misura quantitativa nota come "docking score". Tale punteggio rappresenta la forza dell'interazione prevista, tenendo conto di fattori come interazioni elettrostatiche, legami idrogeno e interazioni idrofobiche. Un valore più negativo del docking score indica tipicamente un'interazione più favorevole, suggerendo una maggiore probabilità di efficacia della molecola candidata.

L'applicazione di tecniche di deep learning al molecular docking mira a superare i limiti computazionali dei metodi tradizionali, che spesso richiedono tempi significativi per la simulazione di interazioni complesse. Attraverso modelli di deep learning, diventa possibile predire con elevata accuratezza il docking score, accelerando il processo di screening e

migliorandone la precisione.

### 0.2 Rappresentazione Molecolare

Un elemento chiave nell'utilizzo del deep learning per la drug discovery è la rappresentazione delle molecole in formati idonei per l'elaborazione da parte delle reti neurali. In questo studio, le molecole sono inizialmente descritte mediante la notazione SMILES (Simplified Molecular Input Line Entry System), una stringa lineare che rappresenta la struttura chimica della molecola. Questo formato, pur essendo molto utilizzato in ambito chimico farmaceutico, presenta delle limitazioni per l'elaborazione automatica, in quanto le reti neurali non possono processare direttamente stringhe di testo.

Per esempio:

CN1C(=O)NC2=CC=CC(NC(=O)C(F)C3CC(N)C3)=C21

rappresenta una molecola complessa, codificando informazioni su atomi e legami chimici. Tuttavia, per sfruttare appieno il potenziale delle reti neurali, queste stringhe SMILES vengono trasformate in rappresentazioni più strutturate utilizzando la libreria RDKit:

- **Molecular Fingerprints:** Sequenze binarie che codificano la presenza di specifiche caratteristiche strutturali nella molecola, come sottostrutture e gruppi funzionali. Questi fingerprint permettono di sintetizzare informazioni complesse in un formato compatto, elaborabile dalle Convolutional Neural Networks (CNN).
- **Molecular Graphs:** Rappresentazioni grafiche in cui gli atomi sono nodi e i legami chimici sono archi, preservando la struttura topologica delle molecole. Questa rappresentazione è particolarmente utile per reti neurali progettate per dati strutturati, come le Graph Neural Networks (GNN).

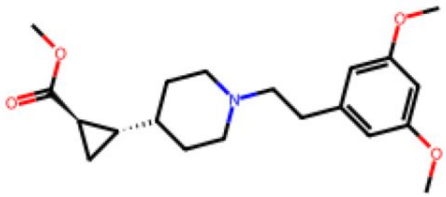

**Figura 1.** Esempio di una formula di una struttura chimica di una molecola organica.

Queste rappresentazioni consentono di catturare informazioni fondamentali per i modelli di deep learning, fornendo una base solida per analisi predittive accurate.

### 0.3 Approccio Deep Learning

Questo studio sfrutta due diverse architetture di deep learning per la predizione del docking score:

- **Convolutional Neural Networks (CNN):** Applicate ai molecular fingerprints. Le CNN trattano queste rappresentazioni come vettori monodimensionali, sfruttando pattern spaziali per estrarre informazioni rilevanti. Grazie alla loro capacità di individuare correlazioni locali nei dati, risultano particolarmente adatte per analizzare caratteristiche molecolari rappresentate come sequenze binarie.
- **Graph Neural Networks (GNN):** Progettate per elaborare i molecular graphs. Le GNN sono in grado di modellare sia le caratteristiche atomiche che le relazioni topologiche tra gli atomi. Queste reti sfruttano la struttura intrinseca dei grafi per apprendere rappresentazioni avanzate che combinano informazioni locali e globali.

L'integrazione di queste due architetture permette di confrontare e combinare approcci diversi, ottimizzando la capacità predittiva del modello. Questo approccio consente inoltre di esplorare quale rappresentazione molecolare (fingerprint o molecular graphs) sia più efficace per il task specifico.

### 0.4 Obiettivo dello Studio

L'obiettivo principale di questo lavoro è sviluppare modelli di deep learning in grado di predire accuratamente i docking score, a partire da rappresentazioni molecolari diverse. Questo approccio mira a:

- Ridurre i tempi e i costi del processo di screening virtuale, migliorando l'efficienza computazionale.
- Fornire uno strumento computazionale rapido per la valutazione preliminare di grandi librerie molecolari, aumentando la probabilità di successo nella selezione dei candidati più promettenti.
- Confrontare l'efficacia di diverse rappresentazioni molecolari e architetture di rete neurale, fornendo indicazioni utili per futuri sviluppi.

Il successo in questa predizione potrebbe trasformare il panorama della drug discovery, accelerando la selezione delle molecole più promettenti e ottimizzando l'utilizzo delle risorse computazionali.

## 1. Metodi

La pipeline del progetto è stata sviluppata con un approccio metodico per la predizione del docking score, integrando due approcci paralleli basati su Convolutional Neural Networks e Graph Neural Networks.

Entrambe le metodologie seguono le stesse fasi principali:

- Preprocessing dei Dati.
- Engineering delle Feature.
- Progettazione delle Architetture di Deep Learning.
- Addestramento dei Modelli.

Questa struttura ha permesso di sfruttare le potenzialità delle CNN, ideali per elaborare rappresentazioni vettoriali come i fingerprint molecolari, e delle GNN, progettate per modellare i dati strutturati come i grafi molecolari. Ogni fase è stata progettata e ottimizzata per garantire robustezza, generalizzazione e confronto diretto tra i due approcci, con l'obiettivo di valutare quale rappresentazione e architettura si dimostri più efficace per il task specifico.

## 1.1 Preprocessing dei Dati

Il dataset iniziale comprende oltre 226.000 coppie di dati, ciascuna composta da una rappresentazione SMILES di una molecola e il suo docking score.

### 1.1.1 Approccio basato su CNN

Per garantire la qualità dei dati destinati alla rete CNN, sono stati implementati i seguenti passaggi:

- **Rimozione dei duplicati:** I fingerprint molecolari generati tramite l'algoritmo Extended Connectivity Fingerprint (ECFP), noto anche come Morgan, sono stati utilizzati per l'identificazione e la rimozione dei duplicati nel dataset. In questo studio è stata adottata una configurazione con *bits* pari a 2048 e *raggio* 2, parametri scelti per ottimizzare il compromesso tra dettaglio della rappresentazione molecolare e costo computazionale. L'elevata dimensionalità del fingerprint scelto consente una discriminazione efficace tra strutture molecolari simili, minimizzando il rischio di falsi positivi nell'identificazione dei duplicati e prevenendo potenziali problematiche di *data leakage*. L'algoritmo Morgan presenta diverse varianti parametriche, che si differenziano principalmente per il numero di bit utilizzati nella codifica e per il raggio di analisi degli intorni atomici. Mentre algoritmi alternativi come MACCS e Daylight si basano rispettivamente su un set predefinito di sottostrutture o su rappresentazioni topologiche, Morgan è stato selezionato per questo studio grazie alla sua flessibilità parametrica, all'esteso supporto nelle librerie di cheminformatica e alla documentata efficacia nella caratterizzazione di strutture molecolari complesse. Per quanto riguarda la percentuale di duplicati presenti nel dataset, l'analisi ha rilevato un valore pari al 19,64%. In presenza di quest'ultimi, è stata mantenuta arbitrariamente la prima occorrenza di ciascuna molecola, rimuovendo le successive.

- **Validazione delle stringhe SMILES:** Con l'aiuto della libreria RDKit, ogni SMILES è stato validato per assicurarsi che fosse convertibile in rappresentazioni molecolari. Nessuna entry è stata rimossa per errori di parsing o invalidità.

### 1.1.2 Approccio basato su GNN

Per garantire la qualità dei dati destinati alla rete GNN, sono stati implementati i seguenti passaggi:

- **Rimozione dei grafi duplicati:** Per garantire la qualità del dataset e prevenire problemi di *data leakage*, è stata implementata una funzione dedicata per eliminare i duplicati. Ogni grafo è stato rappresentato come una tupla immutabile, contenente i tensori delle caratteristiche dei nodi e le informazioni topologiche degli archi. Questa rappresentazione ha permesso di confrontare efficacemente i grafi tra loro. Il processo ha identificato e rimosso i duplicati mantenendo solo la prima occorrenza di ciascun grafo. In questo studio, la rimozione dei duplicati ha portato alla riduzione del dataset originale, migliorando la qualità dei dati e ottimizzando l'efficacia della rete GNN. L'analisi ha rivelato che il 10.84% delle entry nel dataset erano duplicate.

- **Creazione dei grafi molecolari:** Ogni molecola è stata rappresentata come un grafo, dove i nodi corrispondono agli atomi e gli archi ai legami chimici. Utilizzando la libreria RDKit, le stringhe SMILES sono state convertite in oggetti molecolari, da cui sono state estratte le informazioni strutturali. Per ogni atomo, sono stati generati nodi con attributi specifici come numero atomico, grado di connessione, carica formale, tipo di ibridazione e aromaticità. I legami chimici tra gli atomi sono stati rappresentati come archi non direzionati, preservando la topologia molecolare. La funzione di conversione da SMILES a grafo ha garantito l'inclusione di tutti i dettagli molecolari rilevanti. In caso di molecole non valide, queste sono state automaticamente escluse dal dataset per evitare inconsistenze durante l'elaborazione successiva.

- **Codifica delle caratteristiche atomiche e di legame:** Le caratteristiche atomiche sono state codificate in tensori numerici utilizzabili dalla rete. Ad esempio, il numero atomico e il grado di connessione sono stati rappresentati come valori numerici, mentre l'aromaticità e il tipo di ibridazione sono stati codificati come interi. Per quanto riguarda i legami chimici, ogni arco è stato arricchito con attributi quali il tipo di legame (singolo, doppio, triplo) e l'aromaticità del legame stesso. Questo processo ha permesso di creare una rappresentazione grafica dettagliata per ciascuna molecola, catturando informazioni sia locali (caratteristiche degli atomi) che globali (relazioni topologiche tra atomi). Una volta generati, i grafi molecolari sono stati memorizzati come oggetti della libreria PyTorch Geometric.

Le differenze nelle percentuali di duplicati tra i modelli basati su CNN e GNN derivano principalmente dalla rappresentazione molecolare e dai criteri di identificazione dei duplicati. Nel caso delle CNN, la deduplicazione avviene utilizzando fingerprint molecolari generati tramite algoritmi come il Morgan Fingerprint, che sintetizzano la struttura chimica in una rappresentazione binaria standardizzata. Questo approccio semplifica la molecola, ma può trascurare dettagli atomici e strutturali più complessi, portando a una maggiore

probabilità di identificare duplicati, anche tra molecole che sono chimicamente distinte. Al contrario, le GNN utilizzano una rappresentazione basata su grafi molecolari, dove ogni molecola è tradotta in nodi (atomi) e bordi (legami) con attributi specifici. La deduplicazione in questo caso si basa su confronti dettagliati tra grafi, che includono informazioni atomiche e legami. Tuttavia, questa granularità può anche introdurre sensibilità all'ordine atomico e alla struttura, causando una maggiore variabilità nella rilevazione dei duplicati.

## 1.2 Feature Engineering

### 1.2.1 CNN

L'approccio utilizzato si basa su una rappresentazione vettoriale delle molecole tramite fingerprint, elaborati dal modello. Nello specifico, i Morgan fingerprints, generati durante il preprocessing, sono stati ulteriormente trattati attraverso diversi passaggi chiave.

Inizialmente, i dati sono stati mescolati per eliminare qualsiasi ordinamento intrinseco che potesse introdurre *bias* durante l'addestramento del modello. La randomizzazione è stata effettuata utilizzando un seed predefinito, garantendo così la riproducibilità degli esperimenti. Questo accorgimento è fondamentale per evitare che il modello apprenda correlazioni spurie, ossia schemi dovuti all'ordine dei dati piuttosto che a caratteristiche intrinseche.

Successivamente, è stata applicata una fase di normalizzazione dei docking scores utilizzando la funzione *StandardScaler*, che trasforma i valori in modo tale da avere una media pari a zero e una deviazione standard pari a uno. L'obiettivo di questa operazione era migliorare la stabilità dell'addestramento e facilitare la convergenza degli algoritmi di ottimizzazione. Tuttavia, dopo un'analisi comparativa (visionabile alla sezione 2.5), è emerso che la normalizzazione non apportava un miglioramento significativo alle prestazioni complessive del modello. Pertanto, si è deciso di rimuovere la normalizzazione e utilizzare i dati nella loro forma originale per tutte le successive fasi di addestramento. Questa scelta ha permesso di semplificare il preprocessing senza compromettere le prestazioni dei modelli.

Il dataset è stato in seguito partizionato secondo uno schema di divisione gerarchico: l'80% dei dati è stato allocato al *training set*, mentre il restante 20% è stato ulteriormente suddiviso in *validation set* (66.7%) e *test set* (33.3%). Questa strategia di partizionamento garantisce una valutazione robusta delle prestazioni del modello, mantenendo un subset completamente indipendente per la validazione finale.

Per l'ottimizzazione dell'addestramento e della validazione, i dati sono stati gestiti tramite *DataLoader* che organizzano i campioni in batch di 64 elementi, bilanciando efficacemente stabilità dell'ottimizzazione ed efficienza computazionale, consentendo al modello di aggiornare i pesi in modo graduale e uniforme. Inoltre, è stato applicato un meccanismo di mescolatura casuale ("shuffle") dei batch nell'insieme di training, per prevenire adattamenti del modello a particolari sequenze nei dati, mentre i set di validation e test sono stati mantenuti ordinati per garantire una valutazione consistente e riproducibile.

### 1.2.2 GNN

Per la parte inerente alla gestione dei dati nella GNN sono stati seguiti principi simili a quelli applicati per la CNN, con alcune specificità dovute alla struttura grafo-based dei dati. Anche in questo caso, il dataset è stato mescolato utilizzando un seed predefinito per garantire la riproducibilità e prevenire l'introduzione di *bias* legati all'ordinamento intrinseco dei campioni. Questo passaggio è stato fondamentale per assicurare che il modello apprendesse correlazioni significative, basate esclusivamente sulle proprietà strutturali dei grafi.

Per migliorare la stabilità del processo di addestramento, i *docking scores* associati ai grafi sono stati normalizzati utilizzando *StandardScaler*, analogamente a quanto fatto per la CNN.

La suddivisione del dataset è stata eseguita secondo lo stesso schema adottato per le CNN, con una ripartizione gerarchica che prevede un 80% di dati destinato all'addestramento e il restante 20% suddiviso in *validation set* e *test set*.

Durante l'addestramento, i dati sono stati organizzati in batch di dimensione pari a 64, con un meccanismo di mescolatura casuale applicato ai dati del *training set*, così da evitare adattamenti del modello a sequenze specifiche. Per i *validation set* e i *test set*, invece, è stato mantenuto l'ordine dei campioni, al fine di garantire una valutazione consistente e riproducibile.

## 1.3 Progettazione delle Architetture

### 1.3.1 CNN

Questa architettura sfrutta una gerarchia di strati convoluzionali e densi per catturare progressivamente informazioni di alto livello dai dati molecolari. La combinazione di convoluzioni, pooling e strati completamente connessi consente alla rete di apprendere rappresentazioni complesse e precise, essenziali per prevedere accuratamente il docking score. Le tecniche di regolarizzazione, come dropout e batch normalization, sono state integrate per migliorare la generalizzazione del modello e ridurre il rischio di overfitting.

#### Struttura della Rete

La rete neurale convoluzionale progettata per questo studio si compone di tre blocchi convoluzionali seguiti da livelli completamente connessi. Di seguito viene descritta la struttura architetturale della rete:

**Livelli Convoluzionali**

- **Primo livello convoluzionale:**
  - Filtro convoluzionale 1D con **64 canali di output** e un **kernel** di dimensione 3.
  - *Batch Normalization* per stabilizzare la distribuzione delle attivazioni.
  - Funzione di attivazione *ReLU* per introdurre non linearità.

  - *Max-Pooling* con finestra di dimensione 2 per ridurre la dimensionalità delle feature map.

- **Secondo livello convoluzionale:**
  - Filtro convoluzionale 1D con **128 canali di output** e un **kernel** di dimensione 3.
  - Seguito da *Batch Normalization*, *ReLU* e *Max-Pooling* con configurazioni simili al livello precedente.

- **Terzo livello convoluzionale:**
  - Filtro convoluzionale 1D con **128 canali di output** e un **kernel** di dimensione 3.
  - Seguito da *Batch Normalization*, *ReLU* e *Max-Pooling*.

I livelli convoluzionali estraggono caratteristiche rilevanti dai fingerprint molecolari sfruttando convoluzioni locali, seguite da operazioni di pooling per ridurre progressivamente la dimensionalità delle feature map.

#### Livelli Fully Connected (Completamente Connessi)

- **Primo livello fully connected:**
  - Livello lineare con **256 unità**.
  - Funzione di attivazione *ReLU*.
  - *Dropout* con probabilità del 50% per migliorare la generalizzazione.

- **Secondo livello fully connected:**
  - Livello lineare con **64 unità**.
  - Funzione di attivazione *ReLU*.
  - *Dropout* con probabilità del 50%.

- **Livello di output:**
  - Livello lineare con una **singola unità** per fornire la previsione finale del docking score.

### 1.3.2 GNN

Questa architettura utilizza una combinazione di strati convoluzionali specifici per grafi e strati completamente connessi per catturare sia informazioni locali che globali dai grafi molecolari. La rete è progettata per integrare le caratteristiche dei nodi e degli archi, permettendo una rappresentazione efficace delle strutture molecolari. Tecniche di regolarizzazione come il *dropout* e la *batch normalization* sono integrate per migliorare la generalizzazione e ridurre il rischio di overfitting.

## Struttura della Rete

La rete GNN progettata si basa su un'architettura flessibile che combina due blocchi convoluzionali sui grafi con strati di elaborazione per gli attributi degli archi, pooling globale e livelli completamente connessi.

#### Parametri della Rete

- **Caratteristiche dei nodi:** 5 caratteristiche per ogni nodo (***num_node_features = 5***).
- **Caratteristiche degli archi:** 2 attributi per ogni legame (***edge_dim = 2***).
- **Dimensioni nascoste:** 256 unità nei livelli convoluzionali e completamente connessi (***hidden_channels = 256***).
- **Dropout:** Probabilità di **30%** per ridurre il rischio di overfitting.

#### Livelli Convoluzionali per Grafi

- **Primo livello convoluzionale:**
  - Strato *GCNConv* con **256 canali di output**.
  - *Batch Normalization* per stabilizzare la distribuzione delle attivazioni.
  - Funzione di attivazione *ReLU* per introdurre non linearità.
  - *Dropout* con probabilità del **30%**.

- **Secondo livello convoluzionale:**
  - Strato *GCNConv* con **256 canali di output**.
  - Seguito da *Batch Normalization*, *ReLU* e *Dropout* come nel livello precedente.

I livelli convoluzionali lavorano congiuntamente per estrarre caratteristiche dai nodi del grafo e sfruttare le connessioni tra i nodi attraverso gli archi.

#### Elaborazione degli Attributi degli Archi

- Gli attributi degli archi (*edge_attr*) vengono elaborati tramite uno strato lineare (***edge_mlp***) con **256 unità di output**.
- Funzione di attivazione *ReLU* per trasformare gli attributi degli archi in una rappresentazione più utile.

#### Pooling Globale

- *Global Mean Pooling* per aggregare le rappresentazioni dei nodi in una singola rappresentazione globale del grafo.

#### Livelli Fully Connected (Completamente Connessi)

- **Primo livello fully connected:**
  - Livello lineare con **256 unità**.
  - Funzione di attivazione *ReLU*.
  - *Batch Normalization* per stabilizzare le attivazioni.
  - *Dropout* con probabilità del **30%**.

- **Livello di output:**
  - Livello lineare con una **singola unità** per fornire la previsione finale del docking score.

### 1.4 Metriche di Valutazione

La valutazione delle prestazioni dei modelli è stata condotta utilizzando un insieme di metriche standard per problemi di regressione, tra cui *Mean Squared Error* (MSE), *Mean Absolute Error* (MAE) e il *coefficiente di determinazione R-squared* ($R^2$).

La **MSE** è stata impiegata come funzione di perdita durante l'addestramento per minimizzare l'errore quadratico medio tra le predizioni del modello e i valori reali, penalizzando maggiormente gli errori di entità elevata.

La **MAE**, calcolata come media degli errori assoluti, è stata adottata come metrica complementare per fornire una valutazione più robusta agli outlier.

Infine, il **coefficiente di determinazione $R^2$** è stato utilizzato per quantificare la proporzione di variabilità nei dati osservati che il modello riesce a spiegare. Questa metrica fornisce un'indicazione globale dell'accuratezza predittiva.

### 1.5 Approccio di Valutazione

L'approccio seguito per la valutazione delle prestazioni si basa su una combinazione di monitoraggio durante l'addestramento e valutazione post-addestramento su set di validazione e test.

#### Durante l'addestramento:

- La **MSE** è stata utilizzata per ottimizzare i pesi del modello tramite l'algoritmo *Adam*, con il suo valore medio calcolato dopo ogni epoca per monitorare la convergenza.

#### Valutazione su Validation e Test Set:

- Alla fine di ogni epoca, il modello è stato valutato sui set di validazione e test tramite **MSE**, **MAE** e **$R^2$**.
- Le metriche sono state calcolate utilizzando una funzione dedicata che restituisce i tre indicatori chiave per valutare le prestazioni del modello.

Questa metodologia ha permesso di ottenere una visione completa delle capacità predittive del modello, sia durante l'addestramento che nella fase di test, fornendo una misura chiara della sua capacità di generalizzazione su dati non osservati.

### 1.6 Importanza Complessiva delle Metriche

La combinazione di **MSE**, **MAE** e **$R^2$** fornisce un quadro completo delle prestazioni del modello:

- La **MSE** permette di identificare errori significativi
- La **MAE** offre una stima dell'errore medio più robusta rispetto alla sensibilità agli outlier.
- Il **$R^2$** rappresenta un indicatore intuitivo della capacità del modello di spiegare i dati osservati.

Questa combinazione garantisce un'analisi bilanciata ed esaustiva, evidenziando le capacità del modello di predire valori continui con precisione, riducendo al contempo i rischi di sovra-allenamento o underfitting.

### 1.7 Strategie di Regolarizzazione

Per migliorare la generalizzazione e ridurre il rischio di overfitting, sono state implementate le seguenti strategie di regolarizzazione:

- **Dropout:** Una frazione casuale dei neuroni è stata disattivata durante l'addestramento, costringendo il modello a non dipendere da percorsi specifici e migliorandone la capacità di generalizzazione.
- **Batch Normalization:** È stata applicata dopo ciascun livello convoluzionale per stabilizzare le attivazioni, accelerare la convergenza e migliorare la robustezza del modello.
- **Weight Decay:** Introdotto come parametro nell'ottimizzatore *Adam*, il weight decay aggiunge una penalità proporzionale al quadrato della norma dei pesi del modello. Questo aiuta a prevenire che i pesi assumano valori eccessivamente grandi, contribuendo così a ridurre l'overfitting e migliorare la capacità di generalizzazione del modello.

### 1.8 Addestramento dei Modelli

#### Strategie di Addestramento CNN e GNN

L'addestramento dei modelli è stato condotto utilizzando le seguenti configurazioni:

- **Funzione di perdita:** Mean Squared Error (MSE), scelta per penalizzare maggiormente gli errori di predizione significativi, garantendo un apprendimento più preciso
- **Ottimizzatore:** Adam, configurato con un *learning rate* di 0.0001 e un parametro di *weight decay* è pari a 0.0001, per bilanciare efficacemente convergenza e riduzione dell'overfitting.
- **Batch Size:** 64 campioni per batch, scelto per bilanciare stabilità e consumo di memoria.
- **Strategia di early stopping:** L'addestramento è stato interrotto anticipatamente quando la perdita sul validation set non migliorava per 10 epoche consecutive, prevenendo fenomeni di overfitting.

## 2. Risultati

### 2.1 Introduzione ai Risultati

Nell'ambito di questo lavoro, sono stati sperimentati due approcci distinti per la previsione dei valori di *docking score* a partire dagli SMILES: una *Convolutional Neural Network* (CNN) e una *Graph Neural Network* (GNN).

In questa sezione si presentano i risultati ottenuti, mettendo in luce sia i livelli di accuratezza sia la capacità di generalizzazione dei due modelli. Le prestazioni sono state valutate attraverso le metriche descritte in precedenza, consentendo un confronto esaustivo delle potenzialità e delle limitazioni di ciascuna architettura.

## 2.2 Analisi delle Loss per CNN e GNN

Le due immagini mostrano l'andamento della funzione di *loss* sia in fase di addestramento sia in fase di validazione per due diversi approcci di rete neurale:

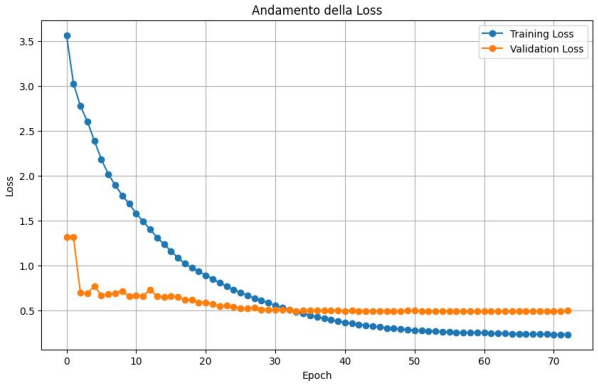


**Figura 2.** Andamento della Loss durante le epoche per la rete CNN.

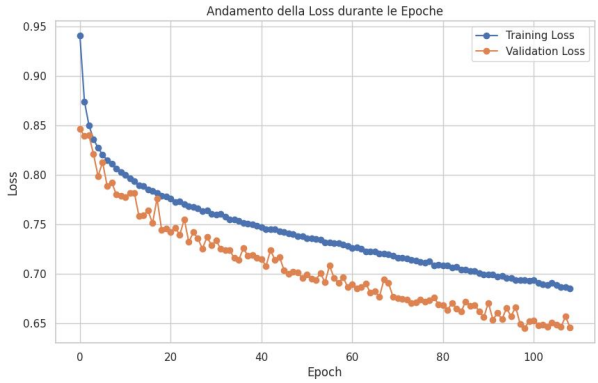


**Figura 3.** Andamento della Loss durante le epoche per la rete GNN.

La Figura 2 evidenzia l'andamento della funzione di costo (*loss*) per la rete CNN, mettendo in luce una rapida discesa della *training loss* durante le prime epoche. Tale comportamento suggerisce che il modello, basato sui fingerprint molecolari, sia in grado di estrarre e apprendere con efficacia le caratteristiche rilevanti dai dati di addestramento. Parallelamente, la *validation loss* si stabilizza in tempi relativamente brevi, indicando un buon livello di generalizzazione già nelle fasi iniziali del training. Sebbene il valore assoluto di *validation loss* risulti superiore a quello di *training loss*, ciò testimonia un effetto di regolarizzazione positivo, in cui il modello evita di adattarsi eccessivamente ai dati di training pur continuando a mantenere una prestazione stabile.

La Figura 3, dedicata invece alla GNN, mostra una diminuzione più graduale della *training loss*, fenomeno imputabile all'aumentata complessità dei dati rappresentati in forma di grafo molecolare. Come per la CNN, nelle fasi iniziali, la *validation loss* si presenta addirittura inferiore rispetto alla *training loss* per poi seguire un trend in progressivo calo. Questo indica che, nonostante la curva di apprendimento sia meno "ripida" rispetto alla CNN, la GNN riesce comunque a migliorare progressivamente la sua capacità di generalizzazione su molecole molto eterogenee.

Dal confronto tra le due reti emerge come la CNN raggiunga *training loss* più bassi, segnale di un apprendimento rapido e incisivo. D'altra parte, la GNN evidenzia oscillazioni più pronunciate nella *validation loss*, presumibilmente dovute a una minore incidenza del *dropout* (30% contro 50% della CNN) e alla complessità intrinseca nell'elaborazione di dati grafici. Entrambe le reti adottano lo stesso *learning rate* (0.0001) e *weight decay* (1e-4), parametri che favoriscono una discesa controllata della *loss* e contenimento del rischio di overfitting.

In definitiva, le differenze rilevate tra CNN e GNN riflettono sia la diversa natura delle rappresentazioni (fingerprint vs grafi molecolari) sia le peculiari strategie di regolarizzazione e dropout impiegate. Nel contesto di analisi molecolare, la CNN offre un apprendimento più rapido e stabile, mentre la GNN richiede tempi di convergenza maggiori ma garantisce un potenziale più alto nella cattura di relazioni strutturali complesse all'interno delle molecole. Questo confronto sottolinea l'importanza di selezionare l'architettura di rete in funzione dell'eterogeneità dei dati disponibili e degli obiettivi di previsione, bilanciando la necessità di un rapido convergere del modello con la capacità di cogliere le relazioni topologiche più sofisticate.

## 2.3 Analisi delle Metriche di Valutazione sui Dati di Test

I valori finali delle metriche di valutazione sui dati di test per entrambe le reti sono riportati nella seguente tabella:

| | **CNN** | **GNN** |
|---|---|---|
| **Validation Loss** | 0.4872 | 0.6269 |
| **Test Loss** | 0.5003 | 0.6503 |
| **Test MSE** | 0.5003 | 0.6503 |
| **Test MAE** | 0.5406 | 0.6169 |
| $\mathbf{R^2}$ | 0.5053 | 0.3649 |

**Tabella 1.** Confronto delle metriche sui dati di test per CNN e GNN.

I valori finali, riportati nella Tabella 1, indicano che la CNN ha ottenuto prestazioni superiori: *Test MSE* e *Test MAE* più bassi e un $R^2$ più elevato, suggerendo una miglior capacità di catturare le relazioni tra i descrittori chimici e i *docking score*. Al contrario, la GNN, pur sfruttando intrinsecamente le strutture molecolari in forma di grafo, ha mostrato un errore e una *loss* più elevati, testimoniando una minor accuratezza complessiva in questo specifico scenario.

Al fine di ottenere i migliori risultati possibili, sono stati condotti molteplici esperimenti sperimentando diverse combinazioni di *learning rate*, valori di *dropout* e modifiche architetturali alle reti neurali. Tali variazioni miravano a migliorare la capacità di generalizzazione e a ridurre ulteriormente gli errori di previsione. Tuttavia, l'ottimizzazione è stata parzialmente limitata dalle risorse computazionali messe a disposizione da Google Colab, le cui sessioni gratuite non consentono di

estendere illimitatamente il numero di epoche di training o di aumentare eccessivamente la complessità del modello. Nonostante tali vincoli, i risultati ottenuti evidenziano la validità dell'approccio proposto e pongono le basi per futuri sviluppi, in cui l'utilizzo di infrastrutture computazionali più potenti potrebbe consentire ulteriori perfezionamenti delle architetture e dell'insieme di iperparametri.

## 2.4 Visualizzazione delle Metriche

Qui di seguito vengono riportati i grafici delle metriche (MSE, MAE e $R^2$) durante le epoche mostrando le differenze tra le due reti:

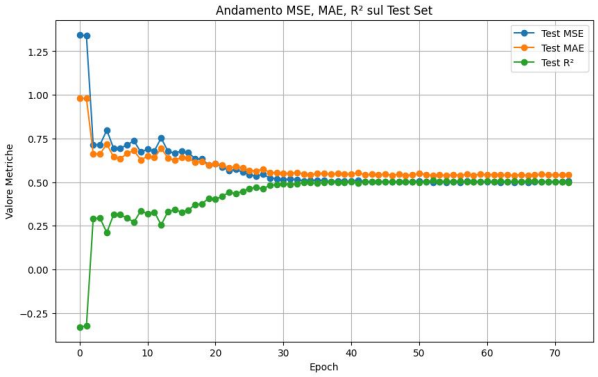


**Figura 4.** Andamento delle metriche (MSE, MAE, $R^2$) sul test set per la rete CNN.

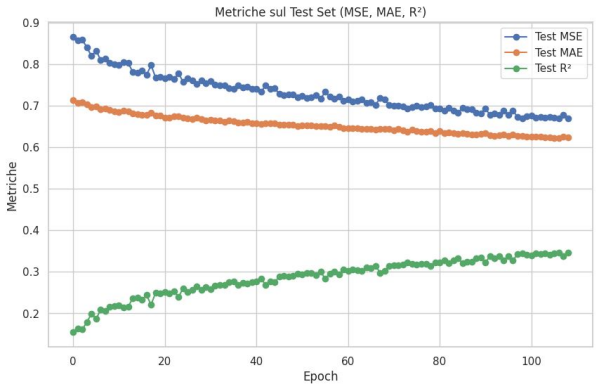


**Figura 5.** Andamento delle metriche (MSE, MAE, $R^2$) sul test set per la rete GNN.

Nel caso della rete CNN (Figura 4), si osserva una diminuzione graduale e relativamente stabile di MSE e MAE durante l'addestramento, a testimonianza della progressiva riduzione degli errori di predizione. Contestualmente, il coefficiente di determinazione mostra un incremento continuo, suggerendo un miglioramento costante nella capacità del modello di spiegare la varianza dei *docking score*. Questo comportamento indica un apprendimento consistente e una convergenza affidabile verso valori ottimali.

Per la GNN (Figura 5), MSE e MAE subiscono invece oscillazioni più pronunciate, soprattutto nelle prime fasi di addestramento, prima di raggiungere una parziale stabilizzazione nelle epoche successive. Allo stesso tempo, i valori di $R^2$ aumentano in modo meno marcato rispetto alla CNN, evidenziando una limitata capacità di catturare le relazioni sottostanti tra le strutture grafiche complesse e i *docking score*. Tale comportamento potrebbe essere dovuto alla natura intrinsecamente più articolata dei grafi molecolari e alla necessità di una più fine sintonizzazione dei parametri di rete per sfruttarne appieno la ricchezza informativa.

Complessivamente, la comparazione tra le due figure mette in luce un vantaggio della CNN nel mantenere errori di predizione inferiori (MSE e MAE più bassi) e un $R^2$ più alto in maniera consistente nel corso dell'allenamento. Nondimeno, la GNN rivela un potenziale interessante, specialmente in scenari in cui le relazioni topologiche nella molecola abbiano un ruolo fondamentale; l'aumento del dataset o l'introduzione di architetture grafiche più sofisticate potrebbe permetterle di ridurre gli sbalzi iniziali e di migliorare la capacità di generalizzazione nel lungo termine.

## 2.5 Impatto della Normalizzazione dei Dati sulle Prestazioni dei Modelli

Le prestazioni dei modelli presentate fino a questo punto sono state ottenute considerando approcci differenti per il preprocessing dei dati:

- CNN: i dati non sono stati normalizzati, mantenedo le caratteristiche originali dei molecular fingerprints.
- GNN: i dati sono stati normalizzati utilizzando una trasformazione tramite *StandardScaler*.

Per valutare l'impatto della normalizzazione sulle prestazioni dei modelli, sono stati eseguiti esperimenti aggiuntivi in cui:

- I dati della CNN sono stati normalizzati utilizzando lo stesso approccio adottato per la GNN.
- I dati della GNN sono stati mantenuti nella loro forma non normalizzata.

I risultati ottenuti sono riassunti nelle seguenti tabelle:

| | **Validation** | **Test** |
|---|---|---|
| **Loss** | 0.6216 | 0.5884 |
| **MSE** | 0.6216 | 0.5884 |
| **MAE** | 0.6076 | 0.59 |
| **$R^2$** | 0.375 | 0.3975 |

**Tabella 2.** Modello CNN con dati normalizzati

| | **Validation** | **Test** |
|---|---|---|
| **Loss** | 0.8097 | 0.8054 |
| **MSE** | 0.8097 | 0.8054 |
| **MAE** | 0.6965 | 0.6928 |
| **$R^2$** | 0.217 | 0.2157 |

**Tabella 3.** Modello GNN con dati non normalizzati

Dai risultati emerge che:

1. Per la CNN, la normalizzazione non apporta miglioramenti significativi alle prestazioni. Al contrario, si osserva un lieve peggioramento delle metriche (MSE, MAE e $R^2$), suggerendo che i molecular fingerprints non necessitano di una trasformazione lineare per ottimizzare l'apprendimento del modello.

2. Per la GNN, mantenere i dati non normalizzati comporta un peggioramento significativo della capacità predittiva, con un aumento di Test Loss e MSE. Questo indica che la normalizzazione è essenziale per garantire una convergenza stabile ed efficace durante l'addestramento della rete.

Questi risultati sottolineano l'importanza di adattare le strategie di preprocessing ai dati e al tipo di architettura utilizzata.

## 2.6 Analisi Critica dei Risultati

Il confronto tra i due approcci di deep learning adottati nello studio mette in evidenza caratteristiche specifiche di ciascun modello, sia in termini di potenzialità che di limitazioni. Sebbene entrambi abbiano mostrato buoni risultati, si vuole in questa sezione analizzare i rispettivi punti di forza e debolezza per comprendere a fondo le implicazioni dei risultati ottenuti e delineare possibili direzioni per il miglioramento futuro.

### Confronto delle Metriche di Valutazione

Dal confronto delle metriche sui dati di test emerge chiaramente che la CNN ha ottenuto performance migliori rispetto alla GNN in termini di *Mean Squared Error* (MSE), *Mean Absolute Error* (MAE) e *coefficiente di determinazione* ($R^2$). Questi risultati suggeriscono una maggiore accuratezza della CNN ed una adattabilità migliore a catturare le relazioni tra le caratteristiche molecolari codificate nei molecular fingerprints e i docking score. Tale risultato riflette la capacità della CNN di catturare con efficienza i pattern chimici essenziali a partire dai fingerprint molecolari.

La GNN, d'altro canto, ha mostrato valori di perdita e di errore leggermente più alti, indicando una minore accuratezza nella predizione dei docking score. Questo risultato si riflette anche in un coefficiente di determinazione ($R^2$) più contenuto, segno che l'architettura grafica, pur essendo più espressiva in teoria, non è ancora riuscita a sfruttare appieno le relazioni topologiche complesse tra gli atomi. Tale limite potrebbe essere attribuito a diversi fattori, tra cui:

- **Complessità della rappresentazione grafica**: La rappresentazione molecolare come grafo, sebbene molto dettagliata e ricca di informazioni, può essere più difficile da elaborare efficacemente per il modello. Le relazioni topologiche complesse tra gli atomi richiedono un numero maggiore di strati convoluzionali per essere apprese, aumentando il rischio di overfitting o di perdita di informazioni durante il pooling globale.
- **Capacità di generalizzazione**: La GNN potrebbe aver risentito di una minore capacità di generalizzazione rispetto alla CNN, come indicato dal valore inferiore del coefficiente di determinazione ($R^2$). Questo suggerisce che la GNN non è stata in grado di catturare pienamente le variazioni nei dati di test, probabilmente a causa della complessità intrinseca del modello o di un dataset di training non sufficientemente ricco di grafi rappresentativi.
- **Efficienza computazionale**: Sebbene non direttamente riportato nei risultati, l'elaborazione grafica è computazionalmente più intensiva rispetto alla manipolazione di vettori. Questo potrebbe aver influito sull'addestramento e sull'ottimizzazione della rete, limitando il numero di esperimenti eseguibili per la GNN.

Complessivamente, quindi, i risultati dimostrano che la CNN rappresenta la soluzione migliore nel presente contesto, garantendo un migliore equilibrio tra accuratezza predittiva e stabilità di addestramento. Tuttavia, la GNN lascia intravedere un potenziale interessante, specialmente se le relazioni topologiche tra atomi avessero un ruolo cruciale nel migliorare la predittività su dataset più complessi e di dimensioni maggiori.

### Stabilità della Validazione

Un aspetto critico emerso è la maggiore oscillazione della *validation loss* nella GNN rispetto alla CNN. Questo comportamento può suggerire che, pur beneficiando di una rappresentazione più ricca (strutture a grafo), la GNN risulti più sensibile a variazioni nei dati, mostrando una minore stabilità in fase di validazione.

### Implicazioni per il Virtual Screening

L'applicazione pratica di queste architetture al contesto del virtual screening richiede una riflessione critica. Sebbene la CNN abbia dimostrato una maggiore accuratezza predittiva, la GNN offre un approccio complementare che potrebbe rivelarsi più utile in scenari specifici, come la predizione di docking score per molecole con strutture chimiche particolarmente complesse o non ben rappresentate da fingerprints.

Un ulteriore vantaggio delle GNN è la loro capacità di catturare direttamente le informazioni topologiche e strutturali delle molecole, che potrebbero risultare critiche per la predizione di altre proprietà molecolari, come tossicità o interazioni chimiche. In questo contesto, è possibile che l'integrazione di ulteriori dati molecolari, come descrittori chimico-fisici, possa migliorare significativamente le performance delle GNN.

### Limiti dello Studio

Uno dei principali limiti dello studio risiede nella dimensione e nella diversità del dataset utilizzato. Sebbene siano stati adottati approcci rigorosi per il preprocessing e il partizionamento dei dati, una maggiore varietà di molecole e docking score avrebbe potuto garantire una valutazione più completa delle capacità di generalizzazione dei modelli. Inoltre, il dataset è stato progettato specificamente per rappresentare

docking score, limitando la generalizzabilità dei risultati ad altri contesti della chimica computazionale.

Un ulteriore limite riguarda l'utilizzo di parametri architetturali standard per entrambe le reti. Sebbene questi siano stati ottimizzati nel corso dell'addestramento, un'esplorazione più approfondita delle configurazioni dei modelli (ad esempio, il numero di strati convoluzionali, la scelta delle funzioni di attivazione o dei meccanismi di pooling) potrebbe portare a miglioramenti significativi nelle performance, specialmente per la GNN.

Infine, le risorse computazionali limitate messe a disposizione da Google Colab hanno rappresentato un vincolo significativo. La piattaforma, pur essendo uno strumento accessibile e versatile, non consente sessioni prolungate o l'utilizzo di GPU particolarmente potenti, limitando così il numero di epoche di training e la complessità architetturale dei modelli sperimentati. Queste restrizioni hanno condizionato l'ottimizzazione dei modelli e ridotto le possibilità di esplorare configurazioni più avanzate, lasciando spazio a potenziali miglioramenti futuri.

### Prospettive Future

Alla luce dei risultati ottenuti, sono possibili diverse direzioni di miglioramento per il futuro:

- **Integrazione di Descrittori Avanzati**: Incorporare ulteriori informazioni molecolari, come proprietà chimico fisiche o dati sperimentali, per arricchire le rappresentazioni e aumentare la precisione predittiva.
- **Integrazione di approcci ibridi**: Unendo la capacità delle CNN di catturare pattern locali con la capacità delle GNN di modellare relazioni topologiche, sarebbe possibile sviluppare un'architettura ibrida che sfrutti i punti di forza di entrambi i modelli.
- **Espansione del dataset**: L'utilizzo di dataset più ampi e diversificati potrebbe migliorare la generalizzazione dei modelli, specialmente per la GNN. Inoltre, l'integrazione di dati sperimentali aggiuntivi, come misurazioni di binding affinity (forza dell'interazione tra ligando e target biologico) o tossicità, potrebbe ampliare il campo di applicazione dello studio.
- **Ottimizzazione architetturale**: Un'analisi più approfondita delle configurazioni dei modelli, incluse tecniche di ricerca iperparametrica automatizzata, potrebbe portare a ulteriori miglioramenti delle performance.
- **Applicazioni multi-task**: Estendere i modelli a predizioni multi-task, in grado di fornire simultaneamente informazioni su docking score e altre proprietà molecolari, potrebbe migliorare significativamente l'efficienza del processo di screening virtuale.

Queste direzioni promettono di consolidare ulteriormente il ruolo del deep learning nella ricerca farmaceutica, favorendo lo sviluppo di farmaci più efficaci in tempi ridotti e con costi contenuti.

## 3. Conclusioni

Il presente studio ha esplorato l'utilizzo di tecniche di deep learning per la predizione dei docking score di molecole rappresentate tramite SMILES. A tal fine, sono stati progettati e implementati due modelli distinti, una rete convoluzionale (CNN) e una rete per grafi (GNN), capaci di elaborare rispettivamente fingerprint molecolari e rappresentazioni grafiche delle strutture chimiche. Attraverso un processo rigoroso di preprocessing, feature engineering e sperimentazione, si è cercato di ottimizzare le performance di entrambe le architetture, con l'obiettivo di valutare quale approccio risultasse più efficace per il task specifico, ossia la stima accurata dei docking score come misura dell'affinità di legame tra le molecole e i loro target biologici.

I risultati ottenuti indicano chiaramente che la CNN si è dimostrata più performante rispetto alla GNN nella predizione dei docking score, con valori inferiori di MSE e MAE, e un coefficiente di determinazione più elevato. Questo evidenzia una maggiore capacità del modello di catturare le relazioni esistenti tra i fingerprint molecolari e i docking score. Tuttavia, la GNN ha mostrato un potenziale interessante, soprattutto per la sua capacità di modellare relazioni topologiche complesse tra gli atomi di una molecola, un aspetto cruciale in scenari dove la rappresentazione strutturale riveste un ruolo centrale.

Un aspetto di rilievo dello studio è rappresentato dall'analisi dell'impatto della normalizzazione dei dati sulle prestazioni dei modelli. Si è osservato che, mentre la CNN non ha beneficiato significativamente dalla normalizzazione, la GNN ha mostrato un peggioramento delle prestazioni in assenza di tale preprocessing. Questo risultato sottolinea l'importanza di adattare le strategie di preprocessing non solo al dataset, ma anche all'architettura utilizzata, fornendo spunti utili per il design di futuri esperimenti.

Al fine di ottenere i migliori risultati possibili, sono stati condotti molteplici esperimenti, esplorando diverse combinazioni di parametri come il *learning rate*, i valori di *dropout* e le configurazioni architetturali delle reti neurali. Queste variazioni avevano l'obiettivo di migliorare la capacità di generalizzazione dei modelli e di ridurre gli errori di predizione. Tuttavia, l'ottimizzazione è stata limitata dalle risorse computazionali disponibili. Google Colab, utilizzato per la fase sperimentale, pur rappresentando uno strumento potente e accessibile. Nonostante tali vincoli, i risultati ottenuti dimostrano la validità dell'approccio proposto e pongono solide basi per futuri sviluppi.

Una delle principali limitazioni dello studio riguarda la dimensione e la diversità del dataset. Sebbene il preprocessing abbia garantito l'eliminazione di duplicati e anomalie, un dataset più ampio e diversificato potrebbe migliorare ulteriormente la capacità di generalizzazione dei modelli, specialmente per la GNN. Inoltre, l'integrazione di dati aggiuntivi, come misurazioni sperimentali di *binding affinity* o descrittori chimico-fisici, potrebbe arricchire le rappresentazioni molecolari e migliorare le performance predittive.

L'adozione di infrastrutture computazionali più potenti permetterebbe di estendere la fase di training e di sperimentare configurazioni architetturali più complesse, come l'integrazione di approcci ibridi tra CNN e GNN. Questa combinazione potrebbe sfruttare i punti di forza di entrambe le reti, unendo la capacità delle CNN di catturare pattern locali con la capacità delle GNN di modellare relazioni topologiche.

Un ulteriore ambito di sviluppo riguarda l'interpretabilità dei modelli, ovvero la capacità di comprendere e spiegare le decisioni prese dalle reti neurali e quindi consentire di individuare quali caratteristiche dei dati influenzano maggiormente le predizioni del modello, rendendo più trasparente il processo decisionale. Ad esempio, nell'ambito della chimica computazionale, questo potrebbe tradursi nell'identificazione dei pattern strutturali molecolari che determinano docking score elevati.

In conclusione, lo studio conferma l'efficacia delle tecniche di deep learning nella predizione dei docking score e sottolinea l'importanza di un design sperimentale adattivo, capace di ottimizzare le performance in funzione delle caratteristiche dei dati e delle reti neurali. I risultati ottenuti rappresentano un passo significativo verso l'utilizzo di modelli avanzati per la drug discovery, aprendo la strada a ulteriori innovazioni in un campo di crescente rilevanza scientifica e tecnologica.

Pertanto, la scelta tra i due approcci dipende dai requisiti specifici dell'applicazione:

- **Applicazioni dove la precisione è critica:** è preferibile utilizzare la CNN.
- **Applicazioni dove l'informazione strutturale è centrale:** la GNN rappresenta una soluzione adeguata.

## Riferimenti bibliografici


[1] Mulugeta Semework, *Drug discovery and Graph Neural Networks (GNNs): a regression example*, Medium, 2024. Disponibile a: https://medium.com/@mulugetas/drug-discovery-and-graph-neural-networks-gnns-a-regression-example-fc738e0f11f3;

[2] Shuyu Wang, Peng Shan, Yuliang Zhao, Lei Zuo *GanDTI: A multi-task neural network for drug-target interaction prediction*, sciencedirect, 2021. Disponibile a: https://www.sciencedirect.com/science/article/abs/pii/S1476927121000438;

[3] Christel Sirocchi, Federica Biancucci, Muhammad Suffian, Matteo Donati, Stefano Ferretti, Alessandro Bogliolo, Mauro Magnani, Michele Menotta, Sara Montagna, *Predicting metabolic responses in genetic disorders via structural representation in machine learning*, springer nature link, 2024. Disponibile a: https://link.springer.com/article/10.1007/s13748-024-00338-9;

[4] Matthew Ragoza, Joshua Hochuli, Elisa Idrobo, Jocelyn Sunseri, David Ryan Koes, *Protein-Ligand Scoring with Convolutional Neural Networks*, Nation library of medicine, 2017. Disponibile a: https://pmc.ncbi.nlm.nih.gov/articles/PMC5479431/;

[5] Tobias Harren, Torben Gutermuth, Christoph Grebner, Gerhard Hessler, Matthias Rarey, *Modern machine-learning for binding affinity estimation of protein–ligand complexes: Progress, opportunities, and challenges*, Nation library of medicine, Wires 2024. Disponibile a: https://wires.onlinelibrary.wiley.com/doi/10.1002/wcms.1716;

[6] Kevin Crampon, Alexis Giorkallo, Myrtille Deldossi, Stéphanie Baud1, Luiz Angelo Steffenel, *Machine-learning methods forligand–protein molecular docking*, 2021;