GenAI to explain CNN

Riccardo Maria Villaggio

Introduzione

In questo lavoro esploriamo un approccio che integra diverse tecnologie per l'analisi e l'interpretabilità delle reti neurali convoluzionali (CNN). Utilizziamo, infatti, il modello di AI generativa **Stable Diffusion 3.5 Large** per creare un dataset di immagini basate su lemmi estratti da **Core WordNet**. Questo dataset viene poi utilizzato come input per analizzare le attivazioni interne di **ResNet-50** tramite *forward hook*, permettendo di associare i pattern di attivazione a concetti semantici specifici definiti nei synset di WordNet.
La pipeline sviluppata comprende la generazione controllata delle immagini, l'estrazione e la media delle attivazioni sui vari layer della CNN, e infine un'analisi quantitativa e qualitativa delle relazioni tra neuroni e concetti semantici.
Un'ulteriore fase di analisi valuta la presenza di bias semantici nelle predizioni di ResNet-50 utilizzando prompt strutturati in modalità "object-only", "context-only" e "mixed" per un gruppo selezionato di classi di ImageNet.
I risultati mostrano che i layer più profondi della CNN riescono a discriminare efficacemente tra i concetti selezionati, confermando la capacità del modello di rappresentare semantiche complesse. L'analisi del bias, invece, evidenzia una marcata dipendenza dell'architettura dalla presenza esplicita di contesto per aumentare la confidenza che il modello ha nelle sue previsioni.

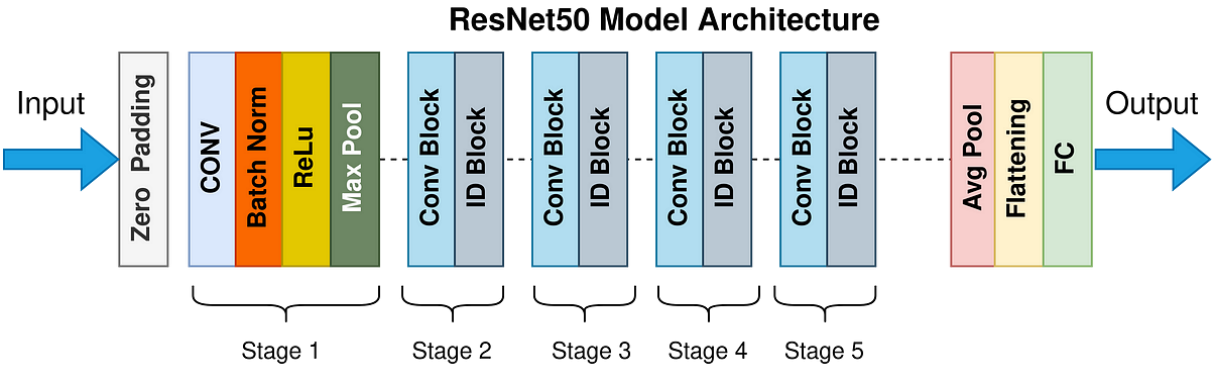

Figure 1: Architettura di *ResNet50*

1. Architettura del Codice

Il progetto è strutturato in cinque classi principali, ognuna con responsabilità ben definite:

1. `CoreWordNetManager`: si occupa del download e parsing del file Core WordNet, della selezione dei synset (distinguendo tra quelli bias-aware e non), della generazione dei prompt sia generici che mirati all'analisi del bias, e della gestione del lessico WordNet.
2. `StableDiffusionGenerator`: gestisce il caricamento del modello Stable Diffusion 3.5 Large e la generazione di batch di immagini a partire dai prompt.
3. `CNNActivationAnalyzer`: carica la rete ResNet-50, esegue il preprocessing delle immagini, estrae le attivazioni medie per layer, calcola metriche utili all'analisi e visualizza i pattern di attivazione.
4. `CNNInterpretabilityProject`: coordina l'intera pipeline di interpretabilità, collegando le attivazioni più significative ai concetti semantici estratti da WordNet e producendo le metriche e visualizzazioni finali.
5. `ImageNetBiasAnalyzer`: gestisce le classi bias, effettua il mapping tra indice di ImageNet e label, identifica i prompt con maggiore probabilità predetta per ciascuna classe bias e produce le metriche specifiche per il report.

2. Spiegazione del Codice

2.1 Configurazione e Parametri (`ProjectConfig`)

Questa classe centralizza tutti i parametri di configurazione: percorsi delle cartelle, nomi dei modelli, dimensioni delle immagini, parametri di generazione e le classi di bias da analizzare.

2.2 CoreWordNetManager

- **Download**: la funzione `_download_and_parse_from_url()` scarica e legge il file Core WordNet, estraendo un set di lemmi.
- **Caricamento synset**: `_load_synsets_from_wordnet()` carica i synset specifici (ad esempio `dog.n.01`), escludendo le classi di bias.
- **Fallback**: se i synset trovati non sono sufficienti, viene utilizzata una lista di parole comuni (utile anche per debug).
- **Prompt CNN-pure**: genera prompt descrittivi utilizzando template predefiniti per ogni categoria semantica.
- **Prompt bias-aware**: genera gruppi di prompt per ciascuna classe bias (`bee`, `frog`, `plane`, `guitar`, `strawberry`), suddivisi in object-only, context-only e mixed.

2.3 StableDiffusionGenerator

- **Caricamento**: `load_model()` tenta prima il caricamento quantizzato a 4 bit, con fallback al modello standard se necessario.
- **Generazione**: `generate_image()` e `generate_batch()` generano immagini a partire dai prompt, salvano i risultati e ottimizzano la gestione della memoria GPU.

2.4 CNNActivationAnalyzer

- **Hook**: registra i *forward hook* sui layer principali di ResNet-50; questi hook catturano le attivazioni durante il forward pass.
- **Analisi**: calcola e stampa diverse metriche utili per l'interpretazione della rete, tra cui:
  - **Attivazione massima per synset e layer**: per ogni layer e synset, viene riportato il valore massimo di attivazione e il numero di immagini considerate.
  - **Correlazioni neurone/synset**: per ciascun layer, vengono mostrate le statistiche (media, deviazione standard, numero di campioni) delle attivazioni dei neuroni più attivi rispetto ai diversi synset.
  - **Separabilità semantica**: viene calcolata la distanza media tra i centroidi delle attivazioni dei synset nei vari layer, utile per valutare la capacità del layer di distinguere tra concetti diversi.

2.5 CNNInterpretabilityProject

Questa classe rappresenta il cuore del progetto e integra tutte le funzionalità principali:

- **Gestione del mapping prompt-synset**
- **Analisi delle metriche prodotte tramite gli hook di ResNet**
- **Esecuzione centralizzata degli esperimenti, facilitando la riproducibilità**
- **Visualizzazione e salvataggio dei risultati**

2.6 ImageNetBiasAnalyzer

La classe `ImageNetBiasAnalyzer` si occupa di:

- Selezionare e gestire le classi bias
- Eseguire la classificazione delle immagini generate per gruppi di prompt (object-only, context-only, mixed)
- Identificare e classificare i prompt in base alla probabilità predetta dalla CNN per ciascuna categoria
- Produrre e salvare le metriche e i report specifici per l'analisi del bias

3. Risultati e Visualizzazioni

Ecco alcuni esempi delle 3 tipologie di immagini generate in base ai prompt adottati:

Figure 2: Esempio di immagine *object only*

Figure 3: Esempio di immagine *context only*

Figure 4: Esempio di immagine *mixed*

Separabilità tra classi nei layer profondi

La heatmap in Figura 5 mostra i valori di attivazione massima per ciascun synset sui migliori 8 layer con maggiore separabilità semantica, ovvero la capacità di distinguere tra i centroidi delle attivazioni delle diverse classi di synsets.

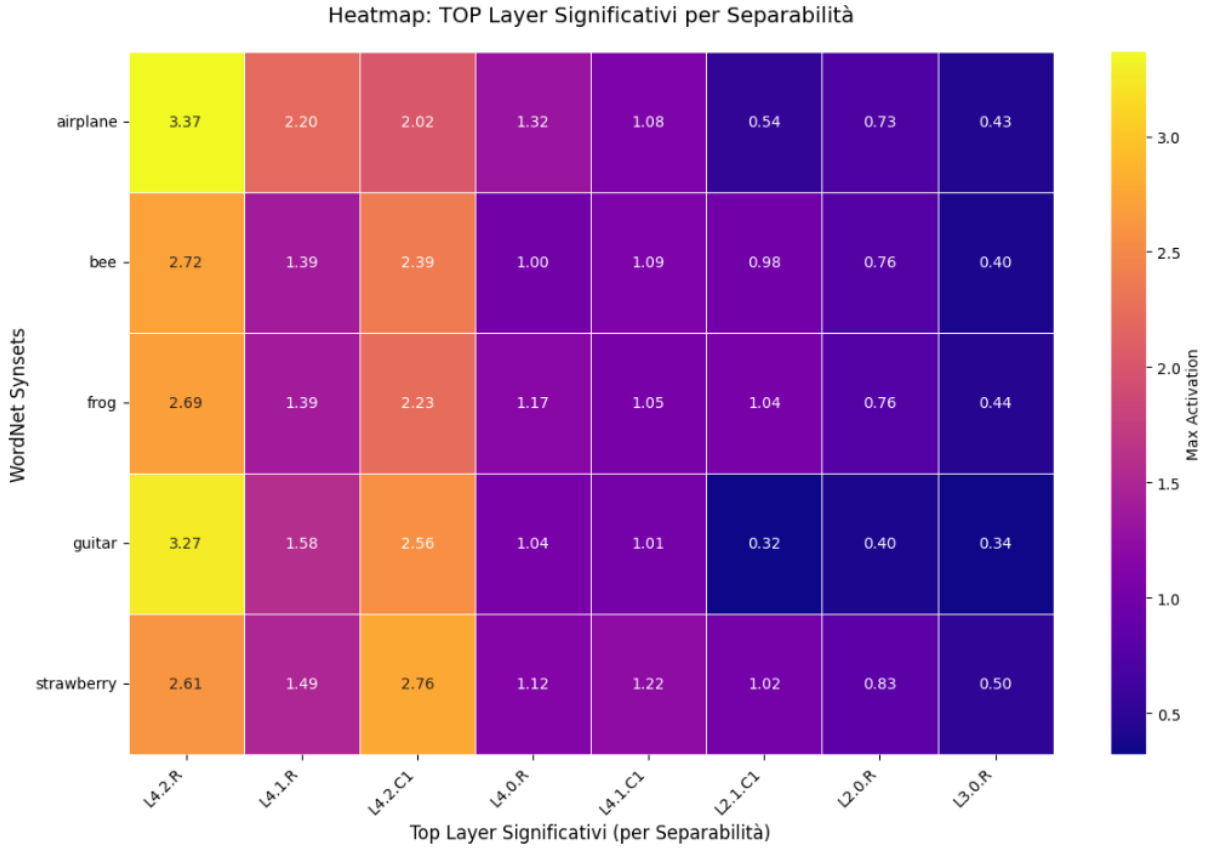

Figure 5: Heatmap per classi di bias sui layer più separabili

Osserviamo:

- `guitar` raggiunge 3.27 su `layer4.2.relu`.
- `airplane` ottiene 3.37 su `layer4.2.relu`, seguito da valori 2.20 e 2.02 per layer precedenti.
- `strawberry` si attesta 2.76 su `layer4.2.conv1` e 2.61 su `layer4.2.relu`.
- `bee` e `frog` presentano 2.72 e 2.69 nei layer top, ma scendono rapidamente.

Risultati quantitativi

Analizzando i migliori 6 layer, si osserva che la separabilità media tra classi — calcolata come distanza media tra i centroidi delle attivazioni dei diversi synset — varia a seconda della profondità del layer: nei layer più profondi si raggiungono valori elevati (ad esempio, 16.828 in layer4.2.relu), a testimonianza della maggiore capacità di distinguere efficacemente i cinque synset testati. È importante sottolineare che la separabilità media globale (cioè tra tutti i centroidi dei synset) risulta sempre superiore rispetto alle attivazioni massime osservate per i singoli synset, poiché la metrica globale tiene conto di tutte le possibili coppie di classi. Ad esempio, la distanza media globale in layer4.2.relu (16.828) è nettamente superiore rispetto all'attivazione massima di synset come airplane o guitar nello stesso layer (rispettivamente 3.37 e 3.27).

Tabella 1: Separabilità media per layer (Top 6)

Layer	**Avg Centroid Distance**
layer4.2.relu	16.828
layer4.1.relu	7.331
layer4.2.conv1	4.443
layer4.0.relu	4.329
layer4.1.conv1	2.952
layer2.1.conv1	2.270

Chiaramenti i valori mostrati sono coerenti a quanto mostrato precedentemente in Figura 5

Tabella 2: Top 5 synset per i 6 layer con maggiore separabilità

Layer	Synset	Max Activation
layer4.2.relu	airplane	3.3669
	guitar	3.2708
	bee	2.7210
	frog	2.6922
	strawberry	2.6121
layer4.1.relu	airplane	2.2011
	guitar	1.5823
	strawberry	1.4935
	bee	1.3906
	frog	1.3889
layer4.2.conv1	strawberry	2.7643
	guitar	2.5606
	bee	2.3852
	frog	2.2322
	airplane	2.0180
layer4.0.relu	airplane	1.3186
	frog	1.1688
	strawberry	1.1203
	guitar	1.0430
	bee	0.9994
layer4.1.conv1	strawberry	1.2223
	bee	1.0875
	airplane	1.0846
	frog	1.0466
	guitar	1.0144
layer2.1.conv1	frog	1.0400
	strawberry	1.0224
	bee	0.9789
	airplane	0.5362
	guitar	0.3222

Confronto con una Esecuzione Precedente (100 prompt)

Un'esecuzione precedente è stata effettuata su un diverso sottoinsieme semantico, con 10 immagini per ciascuno di 10 synset scelti. Di seguito si riportano i risultati nel dettaglio, che mostrano per alcuni layer le classifiche di separabilità per i 3 synsets su cui risultano meglio specializzati (in base alla massima attivazione media):

- `layer4.1.relu`:
  - tree.n.01: 2.5172
  - boat.n.01: 2.3055
  - flower.n.01: 2.1233
- `layer4.0.relu`:
  - flower.n.01: 1.6804
  - boat.n.01: 1.6279
  - computer.n.01: 1.3162

- `layer3.3.relu`:
  - computer.n.01: 0.4060
  - dog.n.01: 0.3881
  - flower.n.01: 0.3645
- `layer2.2.bn2`:
  - computer.n.01: 0.3557
  - car.n.01: 0.3469
  - cat.n.01: 0.3369
- `layer2.1.bn3`:
  - tree.n.01: 0.2985
  - flower.n.01: 0.2932
  - bicycle.n.01: 0.2736
- `layer1.0.bn2`:
  - computer.n.01: 0.5834
  - dog.n.01: 0.5657
  - chair.n.01: 0.5599
- `layer4.0.bn1`:
  - dog.n.01: 0.1840
  - cat.n.01: 0.1813
  - flower.n.01: 0.1747
- `layer4.2.bn1`:
  - bicycle.n.01: 0.2196
  - tree.n.01: 0.2030
  - chair.n.01: 0.1647
- `layer3.5.conv3`:
  - dog.n.01: 0.1595
  - bird.n.01: 0.1476
  - cat.n.01: 0.1155
- `layer4.2.conv2`:
  - chair.n.01: 0.0890
  - cat.n.01: 0.0845
  - dog.n.01: 0.0830

Osservazioni: Anche in questa run si conferma il trend: i layer più profondi (`layer4.*`) mostrano valori di separabilità semantica più elevati, evidenziando una maggiore capacità di discriminare tra concetti anche in presenza di prompt più vari. Le classi maggiormente separabili differiscono leggermente, ma includono concetti diversi come *dog*, *tree*, *flower*, e *boat*.

Esempi di attivazione per singoli neuroni

Per approfondire meglio il comportamento interno della rete, sono state anche analizzate le attivazioni medie dei neuroni in risposta ai prompt generati. Di seguito si mostrano tre esempi significati tratti da `layer4.2.relu`, poichè come visto è il layer più di alto livello e performante per il riconoscimento delle classi, scelti per la loro capacità di evidenziare pattern distintivi rispetto ai diversi synset. Insieme alle medie, si riportano anche le deviazioni standard, che indicano la variabilità dell'attivazione sulle 8 immagini di ciascuna classe.

Tabella 3: Attivazioni medie e deviazioni standard per i neuroni in `layer4.2.relu`

Neurone	Synset	Mean	Std
624	bee	1.569	0.683
	frog	2.616	0.868
	airplane	1.152	1.171
	guitar	2.276	0.710
	strawberry	1.369	0.690
589	bee	1.194	0.607
	frog	1.028	0.606
	airplane	1.487	1.465
	guitar	3.220	1.660
	strawberry	0.518	0.258
617	bee	2.721	1.374
	frog	1.629	1.274
	airplane	0.684	0.278
	guitar	0.487	0.328
	strawberry	1.732	1.211

In questi esempi, si osserva come ogni neurone risponda in modo differenziato alle classi testate. Ad esempio, il neurone 624 è particolarmente attivo per `frog`, mentre il neurone 617 mostra una ampia variabilità su `bee`. Tali pattern indicano la presenza di neuroni specializzati nella rilevazione di caratteristiche visive proprie dei vari synset.

4. Analisi del Bias

I test bias-aware evidenziano una chiara dipendenza del modello dalla presenza esplicita di un contesto visivo all'interno dell'immagine. In particolare, l'aggiunta di elementi riconoscibili e centrali porta sistematicamente a un incremento della confidenza nella classificazione. L'analisi qualitativa dei prompt generati mostra il seguente comportamento:

- I prompt **object-only**, in cui l'oggetto target è presentato senza contesto, generano predizioni corrette con buona confidenza. I valori variano da circa 0.61 (`airplane`) a oltre 0.99 (`strawberry`), confermando che il modello riesce a riconoscere la classe anche in assenza di indizi ambientali. Nostante ciò, è possibile osserva anche l'esatto opposto: per la classe `bee`, `airplane` e `strawberry` alcuni prompt non sono stati correttamente classificati.
- I prompt **context-only**, che rimuovono completamente l'oggetto lasciando solo ambienti semanticamente coerenti, vengono generalmente ignorati: la confidenza è sempre inferiore all'1%, anche in casi fortemente associativi (come stagni per `frog` o piste d'atterraggio per `airplane`). Questo suggerisce che il modello non è affetto da elevato bias nei confronti del contesto.
- I prompt **mixed**, che includono sia l'oggetto che un contesto plausibile, mostrano un comportamento variegato. Per classi come `bee` e `strawberry`, il contesto rafforza la predizione e porta a confidenze prossime al 100%. Al contrario, in classi come `guitar`, l'aggiunta del contesto può talvolta ridurre la confidenza, probabilmente perché l'ambiente può essere fuorviante per la classificazione.

Le tabelle seguenti riportano, per ciascuna classe target, i valori di confidenza ottenuti dal modello sui diversi prompt, suddivisi per categoria:

Table 1: Classe: `bee`

Prompt	**Confidence**
Mixed	
A bee in a beautiful garden setting with colorful flowers and lush greenery.	0.9950
A bee collecting nectar from blooming flowers in a sunny meadow.	0.9805
Object-only	
A bee on a clean white background showing intricate wing and body detail.	0.9735
A professional close-up photograph of a bee with detailed anatomy.	0.1645
A simple illustration of a bee isolated against a plain background.	0.0097
Context-only	
A close-up of pollen-covered blossoms in a sunlit garden.	0.2678
A field of daisies and sunflowers under bright daylight.	0.0047
A vibrant garden full of colorful flowers and greenery.	0.0044

Table 2: Classe: `frog`

Prompt	**Confidence**
Mixed	
A frog resting on lily pads in a serene wetland environment.	0.9020
A frog sitting near a crystal clear pond with lush marsh vegetation.	0.0698
Object-only	
A frog on a clean white background showing intricate skin patterns.	0.9348
A simple illustration of a frog isolated against a plain background.	0.6945
A professional wildlife photograph of a frog with detailed skin texture.	0.5284
Context-only	
A garden bed with green foliage and growing vegetation.	0.0005
A close-up of pollen-covered blossoms in a sunlit garden.	0.0005
A tranquil pond scene with aquatic plants and lily pads.	0.0004

Table 3: Classe: `airplane`

Prompt	**Confidence**
Mixed	
A airplane flying through clear blue sky with puffy white clouds.	0.7531
A airplane approaching the runway at a busy international airport.	0.9531
Object-only	
A airplane on a clean white background showing aerodynamic features.	0.6493
A professional close-up photograph of a airplane with detailed fuselage.	0.4126
A simple illustration of a airplane isolated against a plain background.	0.7717
Context-only	
An airport runway with control tower in the distance.	0.0555
A clear blue sky filled with puffy white clouds.	0.0005
A vast blue sky with scattered cumulus clouds at sunset.	0.0001

Table 4: Classe: `guitar`

Prompt	**Confidence**
Mixed	
A guitar in a recording studio with professional lighting and equipment.	0.9498
A guitar on a concert stage with dramatic stage lights and microphones.	0.0040
Object-only	
A simple illustration of a guitar isolated against a plain background.	0.9789
A guitar on a clean white background showing its shape and strings.	0.9227
A professional product photograph of a guitar with detailed wood grain.	0.0004
Context-only	
A music venue with amplifiers and performance equipment.	0.0090
A recording studio with professional audio equipment and lighting.	0.0030
A concert stage with dramatic stage lights and microphone setup.	0.0143

Table 5: Classe: `strawberry`

Prompt	**Confidence**
Mixed	
A strawberry ripening on its plant surrounded by lush green leaves.	0.9998
A strawberry growing on a healthy green plant with leaves in a garden.	0.9979
Object-only	
A strawberry on a clean white background showing its seeds and texture.	0.9988
A professional macro photograph of a strawberry with detailed surface.	0.9856
A simple illustration of a strawberry isolated against a plain background.	0.3255
Context-only	
A close-up of pollen-covered blossoms in a sunlit garden.	0.0072
A garden bed with green foliage and growing vegetation.	0.0007
A tranquil pond scene with aquatic plants and lily pads.	0.0006

Infine, si osserva che nella classifica dei top prompt per ciascuna classe, può capitare che i prompt context-only presenti siano in realtà prompt pensati per altre classi, ma, nonostante ciò, attivano maggiormente la classe target rispetti a quelli specificatamente progettati per essa. Alla luce di quanto esposto, questo fenomeno conferma che la CNN tende a basare la classificazione anche (e, in alcuni casi, soprattutto) dal contesto su cui è stata allenata a riconoscere la classe target.

5. Conclusioni

- L'integrazione tra WordNet, Stable Diffusion e CNN ha permesso di esplorare in modo innovativo l'interpretabilità delle reti neurali, collegando concetti semantici, generazione di immagini e analisi delle attivazioni interne.
- Le metriche sulle attivazioni e le visualizzazioni ottenute mostrano una buona coerenza tra i concetti semantici di WordNet e le risposte della rete, confermando la validità dell'approccio.
- L'analisi dei bias ha evidenziato come le CNN possano essere influenzate da correlazioni spurie presenti nei dataset di addestramento, sottolineando la necessità di strumenti interpretativi per individuare e mitigare queste vulnerabilità.

Riferimenti

- Salient ImageNet Explorer – University of Maryland:
  `https://salient-imagenet.cs.umd.edu/explore.html`
- Stable Diffusion 3.5 Large – Hugging Face (Stability AI):
  `https://huggingface.co/stabilityai/stable-diffusion-3.5-large`
- Core WordNet Synsets (Princeton University):
  `https://wordnetcode.princeton.edu/standoff-files/core-wordnet.txt`
- *"Core-ImageNet: Measuring Bias in Vision Models Using Core WordNet"*, 2024 – arXiv preprint:
  `https://arxiv.org/html/2412.13079v1`