

Università degli Studi di [...]

Dipartimento di Ingegneria e Scienze Informatiche

# Architettura del Sistema di Captioning SVG

Tesi di Laurea Magistrale in  
Intelligenza Artificiale

<b>Candidato:</b>	Ediluzio
<b>Relatore:</b>	Prof. Nome Cognome
<b>Correlatore:</b>	Dott. Nome Cognome
<b>Anno Accademico:</b>	2023-2024

# Specifiche Tecniche Complessive

Dettagli sull'ambiente di sviluppo e addestramento: [source: 648]

**GPU Utilizzate:** NVIDIA Tesla K80 (24GB VRAM) [source: 648] **Precisione:** Mista FP16/FP32 (abilitata specialmente su GPU K80 per efficienza) [source: 648] **Batch Size Effettivo (con Grad Accum):** Generalmente 16 (es. 8 per device con accumulo 2, o 4 per device con accumulo 4) [source: 648] **Librerie Principali:** PyTorch (~2.0+), Transformers (~4.30+), Datasets, varie per SVG (cairosvg, lxml, ecc.) [source: 648, requirements files]

## 1 Architettura Decoder-Only (Approccio Vettoriale Diretto)

Questa architettura elabora direttamente l'input SVG in formato vettoriale, tokenizzandolo per un modello linguistico decoder-only. [source: 623, 624]

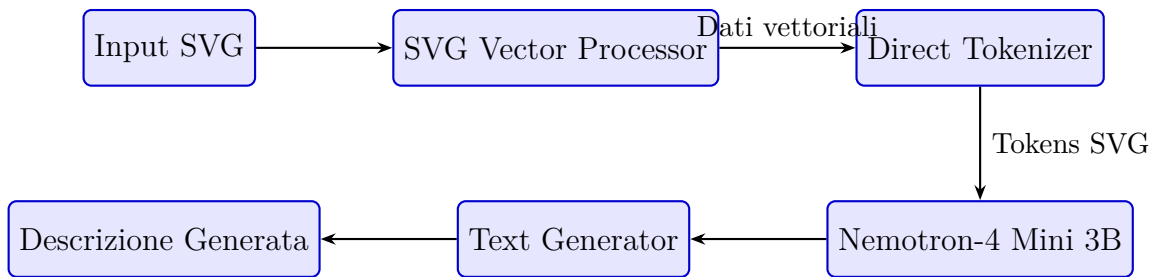


Figura 1: Flusso dell'architettura decoder-only con elaborazione vettoriale diretta. [source: 624]

### Dettagli Implementativi Chiave

**Preprocessing Vettoriale:** Estrazione diretta di comandi path, forme geometriche, attributi e struttura dal codice SVG (`'svg_vector_processor.py'`). [source : 624] **Tokenizzazione Diretta:** `4Mini3B(ovarianteInstruct)utilizzato comodecoderautoregressivo('nemotron_model.py')`. [source : 624] **Generazione Testo:** `Utilizzodibeamsearchocampionamentopergenerareladescrizionefinale('text_generator.py')`. [source : 631]

### Parametri Indicativi (da PDF)

Tabella 1: Parametri indicativi per l'architettura Decoder-Only (basati su PDF forniti). [source: 649]

Componente	Configurazione / Dettaglio	Performance (indicativa)
SVG Vector Processor	Estrazione Vettoriale	-
Tokenizer	Basato su Nemotron + Token SVG	~0.8ms per SVG
Nemotron-4 Mini 3B	~3 Miliardi parametri	~45ms per inferenza

## 2 Architettura Encoder-Decoder

Introduce un encoder specifico per SVG per fornire un contesto più ricco al decoder attraverso meccanismi di cross-attention. [source: 625]

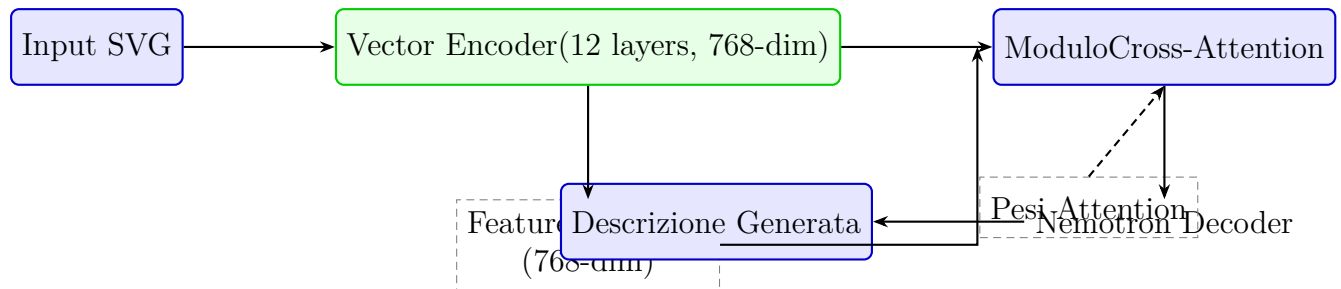


Figura 2: Architettura encoder-decoder con cross-attention e dettagli dimensionali. [source: 625, 626]

### Specifiche Tecniche (da PDF)

Tabella 2: Specifiche indicative per l'architettura Encoder-Decoder. [source: 626]

Componente	Configurazione Indicativa	Note
Encoder	12 layer, 768 hidden dim	Addestrato su ~1M SVG
Cross-Attention	8 heads	Temperatura 0.1 (?)
Decoder	Nemotron-4 Mini 3B	Primi layer congelati durante addestramento iniziale

### Componenti Chiave

- **Encoder Vettoriale:** Modulo dedicato (probabilmente basato su Transformer) per processare la struttura SVG e generare feature vettoriali. [source: 626]
- **Modulo Cross-Attention:** Implementato in `encoder_decoder_conditioning.py` e `encoder_decoder_integrate.py`.