Intelligenza Artificiale e Machine Learning

Presentazione AI/ML

Il Test di Turing

Intelligenza Artificiale

> "Può una macchina pensare?" - Alan Turing, 1950

Il Test di Turing: Se un essere umano non riesce a distinguere tra le risposte di una macchina e quelle di un altro essere umano, allora la macchina può essere considerata "intelligente"

AI Simbolica vs Machine Learning

AI Simbolica

- Regole codificate manualmente
- Logica "se-allora"
- Sistemi esperti
- **Vantaggi**: Interpretabile
- **Svantaggi**: Fragile, rigida

Machine Learning

- Riconoscimento di pattern dai dati
- Reti neurali
- Apprendimento statistico
- **Vantaggi**: Adattabile
- **Svantaggi**: "Scatola nera"

La Guerra delle AI (1960-1990)

- **1960s-1980s**: Dominio dell'AI Simbolica
- **1980s-1990s**: Fallimento dei sistemi esperti
- **Inverno dell'AI**: Perdita di fiducia e finanziamenti
- **Vittoria del ML**: Il machine learning emerge come approccio dominante
- **Oggi**: I Large Language Models dimostrano la supremazia del ML

Machine Learning 1.0

I Fondamenti

Domanda: Qual è la ricchezza di un individuo?

Machine Learning *in nuce*

- La funzione f **riassume** la **relazione** fra fattori $x_1, x_2, ..., x_n$ e y
- L'obiettivo è **predire** y quando abbiamo a disposizione solamente le $x_1, x_2, ..., x_n$
- Vogliamo **imparare** (learn) f per poter poi predire y
- Molti esempi $(y, x_1, x_2, ..., x_n), n \to \infty$ e/o molte variabili $p \to \infty$

Obiettivi vs Sfide

Obiettivo Principale

Generalizzazione: I modelli devono funzionare bene su dati nuovi

Sfide Chiave

- **Overfitting/Underfitting**: Modelli troppo complessi o troppo semplici
- **Limitazioni dei Dati**: Mancanza di dati, dati distorti, dati sbilanciati
- **Complessità Computazionale**: Modelli con molti dati e predittori richiedono ingenti risorse informatiche

---

Complessità e Flessibilità

$$y = f(x_1, x_2, x_3, ..., x_n)$$

La Sfida della Complessità

- Idealmente vorremmo usare una funzione molto complessa che catturi le **interazioni** fra fattori
- La funzione più **flessibile** è una rete neurale (almeno in teoria!)

Il Problema Storico

- Le reti neurali hanno tantissimi parametri e richiedono molti dati e potenza di calcolo
- Fino al 2000: fallimento quasi totale delle reti neurali
- Altri algoritmi (teoricamente meno flessibili) avevano performance migliori

---

La Rinascita delle Reti Neurali

Il Punto di Svolta (2010+)

- **Hardware**: CPU/GPU potenti e accessibili
- **Big Data**: Disponibilità di enormi dataset
- **Algoritmi**: Miglioramenti nell'addestramento

Primo Successo: Riconoscimento Immagini

```
Input: [Immagine di una persona]
Output: "Giuseppe" = f(pixel dell'immagine)
```

Machine Learning 1.0: Caratteristiche

- **Modelli specializzati** per task specifici con ottime performance
- **Problemi teorici** largamente risolti
- **Software** diventato commodity

Chiavi per l'Implementazione Corretta:

- **Gestione dati** (qualità, pulizia, preprocessing)
- **Produzione e manutenzione** (MLOps)
- **Interpretazione** (gestione falsi positivi/negativi)

Machine Learning 2.0

I Modelli di Fondazione

Definizione

I **Foundation Models** sono modelli di grandi dimensioni addestrati su enormi quantità di dati che possono essere adattati per molti compiti diversi.

Caratteristica Principale

"Producono una continuazione *ragionevole* di qualsiasi testo"

Esempio Pratico: GPT in Azione

```python
from openai import OpenAI
import numpy as np
import pandas as pd

client = OpenAI(
    api_key='[omesso]',
    organization='[omesso]',
)

response = client.chat.completions.create(
    model="gpt-3.5-turbo",
    messages=[
        {"role": "user", "content": "La cybersecurity è"},
    ],
    max_tokens=100,
```

```
    temperature=0.2
)
```

Esempio Pratico: Generazione di Testo

```
from openai import OpenAI

client = OpenAI(api_key='[omesso]')

response = client.chat.completions.create(
    model="gpt-3.5-turbo",
    messages=[
        {"role": "user", "content": "La cybersecurity è"},
    ],
    max_tokens=100,
    temperature=0.2
)
```

Input: "La cybersecurity è..."

Output: "La cybersecurity è la pratica di proteggere sistemi informatici, reti, programmi e dati da attacchi digitali..."

💡 Questo funziona! Gli LLM eccellono nella generazione di testo coerente.

Come Funziona? Le Probabilità

Il Processo di Generazione

1. Analizza tutte le occorrenze di "La cybersecurity è la pratica di"
2. Seleziona la continuazione più **frequente** (probabilità più alta)
3. Genera testo basandosi su pattern appresi

Problemi Fondamentali:

- Non ci sono abbastanza frasi che corrispondono esattamente
- Il significato dipende dal **contesto** della frase

Il Problema della Rappresentazione

```
response = client.chat.completions.create(
    model="gpt-3.5-turbo",
    messages=[
```

```
        {"role": "user", "content": "Questa mattina Nagore si è svegliata
molto triste. Era stata presa in giro a scuola e non riusciva a dimenticare
questa esperienza."}
    ]
)
```

Problema: Come mappiamo il testo in numeri?

Soluzione: Gli Embeddings

Definizione Matematica

> In matematica, un embedding è un'istanza di una struttura matematica contenuta all'interno di un'altra istanza.

Esempio Famoso

Italia + Germania - Hitler = Mussolini

```
model.most_similar(positive=["italy", "hitler"],
                   negative=["germany"])
```

Risultato: `[('mussolini', 0.858), ('fascist', 0.712), ...]`

L'Importanza del Contesto

Esempio 1: Ambiguità

```
response = client.chat.completions.create(
    model="gpt-3.5-turbo",
    messages=[{"role": "user", "content": "La regina è stata trovata morta."}]
)
```

Esempio 2: Contesto Chiarificatore

```
response = client.chat.completions.create(
    model="gpt-3.5-turbo",
    messages=[{"role": "user", "content": "L'alveare di mio zio si trova in
fondo alla collina. Purtroppo, la regina è stata trovata morta."}]
)
```

Fine-tuning per Istruzioni

Da GPT a ChatGPT

Adattamento del Modello

Continuare l'addestramento su dataset basati su istruzioni

Obiettivo

Minimizzare la differenza tra l'output del modello e l'output desiderato

Tecniche

- **Apprendimento supervisionato** con dati etichettati
- **Reinforcement Learning from Human Feedback (RLHF)**

Apprendimento Supervisionato: Esempio 1

```
{
  "instruction": "Scrivi una breve storia in terza persona su un protagonista
che deve prendere una decisione importante per la carriera.",
  "input": "",
  "output": "Giovanni si trovava a un bivio della sua vita. Si era appena
laureato e ora doveva affrontare la grande decisione di quale carriera
intraprendere. Dopo molte riflessioni, decise che voleva diventare un
contabile e aiutare i meno abbienti finanziariamente..."
}
```

Formula dell'Addestramento

$$\text{Output Atteso} \approx f(\text{Testo di Input})$$

Apprendimento Supervisionato: Esempio 2

```
{
  "instruction": "Fai il reverse engineering di questo codice per creare una
nuova versione",
  "input": "def fattoriale(num):\n    factorial = 1\n    for i in range(1,
num):\n        factorial *= i\n    return factorial",
  "output": "def fattoriale(num):\n    factorial = 1\n    for i in range(num,
0, -1):\n        factorial *= i\n    return factorial"
}
```

Processo

1. Raccogliere dataset con coppie istruzione-output
2. Addestrare il modello affinché: **Istruzione Attesa** $\approx f(\text{Testo di Output})$

Conclusioni e Futuro

Machine Learning 1.0

- Modelli specializzati per task specifici
- Alta performance su compiti limitati
- Richiede expertise per ogni dominio

Machine Learning 2.0

- Modelli generici adattabili
- Capacità emergenti impreviste
- Verso l'Intelligenza Artificiale Generale (AGI)?

Prossimi Passi

- Multimodalità (testo, immagini, audio, video)
- Reasoning più sofisticato
- Efficienza computazionale
- Allineamento con valori umani

Domande?

Grazie per l'attenzione!

"Il futuro appartiene a coloro che credono nella bellezza dei propri sogni... e sanno programmare."