Analisi Comparativa del Clustering di Risposte LLM con BERT Fine-Tuned e BERT Base

Autore: *Il Tuo Nome*[*]

April 25, 2025

Abstract

In questo lavoro conduciamo un'analisi comparativa tra due pipeline di estrazione e clustering di embedding di risposte generate da LLM. Nel primo esperimento (Run A), utilizziamo un modello *BERT-base* fine-tuned per distinguere risposte "jailbreak" da risposte regolari; nel secondo (Run B), impieghiamo il medesimo modello BERT-base in versione non fine-tuned. Entrambi i set di embedding vengono raggruppati tramite K-Means ($k = 2$) e visualizzati mediante PCA e t-SNE. Confrontiamo distribuzioni, separabilità e compattezza dei cluster per valutare l'effetto del fine-tuning sulla rappresentazione vettoriale.

1 Introduzione

La crescente diffusione di modelli di linguaggio di grandi dimensioni (LLM) impone la necessità di identificare automaticamente risposte indesiderate, ad esempio tentativi di jailbreak. In letteratura, l'addestramento supervisionato di un classificatore su esempi annotati migliora la discriminazione; tuttavia, è interessante capire come il fine-tuning influisca sulla struttura degli embedding stessi. Il presente studio si concentra sul confronto tra embedding estratti da un BERT-base fine-tuned e un BERT-base non fine-tuned, usando clustering non supervisionato.

[*]Email: tuonome@esempio.com

2 Metodologia

2.1 Dataset e Pre-elaborazione

Le risposte LLM analizzate (`response.json`) comprendono sia esempi di jailbreak sia risposte conforme alle policy. Nessuna ulteriore normalizzazione è applicata al testo: ogni risposta è tokenizzata con il tokenizer di BERT.

2.2 Modelli di Embedding

- **Run A (BERT Fine-Tuned):** modello `Teto03/Bert`$_b ase_f ine Tuned(2 classi) caricato tram$

- `Run B (BERT Base): modello pre-addestrato bert-base-uncased, caricato tramite BertModel, senza ulteriori personalizzazioni; estrazione identica del vettore [CLS].`

2.3 Clustering e Visualizzazione

Per entrambi i run, i vettori [CLS] vengono raggruppati con K-Means ($k = 2$, random_state = 42). La separazione dei cluster è rappresentata con:

1. PCA (2 componenti principali) per una panoramica lineare.
2. t-SNE (2D, *perplexity*= $\min(30, n\_samples - 1)$) per un embedding non lineare ad alta risoluzione.

Le figure **??**–**??** presentano rispettivamente PCA e t-SNE per i due esperimenti.

3 Risultati

3.1 Distribuzione dei Cluster

La Tabella **??** riporta la ripartizione delle risposte nei due cluster.

Table 1: Distribuzione dei risposte per cluster nei due esperimenti

Modello	Cluster	Conteggio	Percentuale
BERT Fine-Tuned	0	1087	61.31%
	1	686	38.69%
BERT Base	0	331	18.67%
	1	1442	81.33%

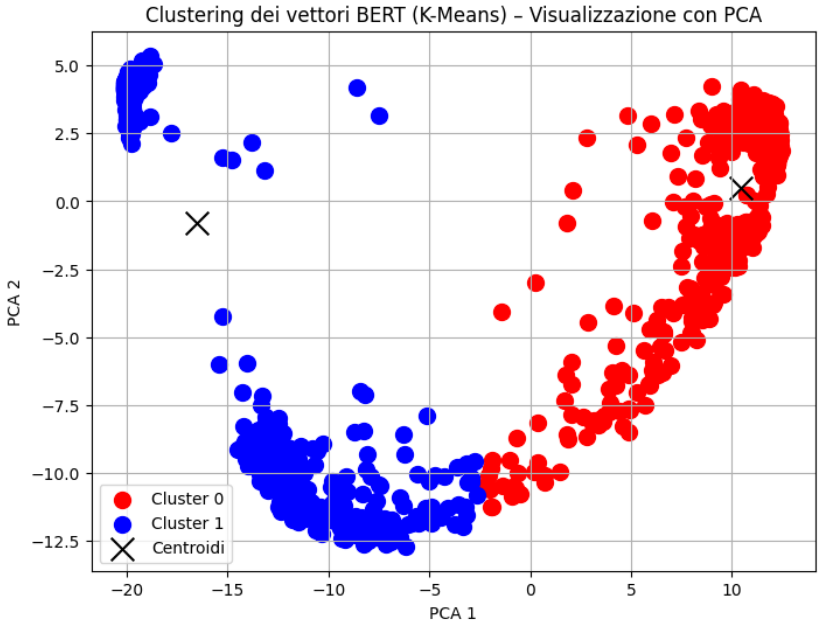

Figure 1: PCA dei cluster estratti dal BERT Fine-Tuned (Run A).

3.2 Visualizzazioni PCA

3.3 Visualizzazioni t-SNE

4 Discussione

L'analisi mostra differenze sostanziali nella distribuzione e nell'organizzazione degli embedding:

- **Distribuzione:** il modello fine-tuned assegna maggiormente le risposte al Cluster 0 (61%), suggerendo che la rete personalizzata riconosce un gruppo consistente di esempi con caratteristiche comuni (presumibilmente jailbreak). Il BERT base invece concentra l'81% nel Cluster 1, evidenziando un bias dell'embedding non addestrato verso una sola re-

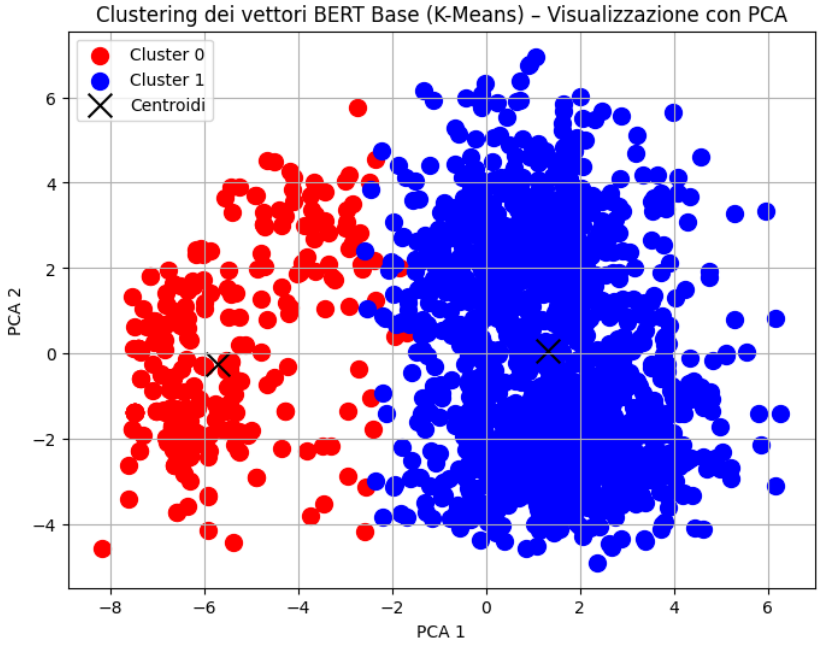

Figure 2: PCA dei cluster estratti dal BERT Base (Run B).

gione dello spazio.

- **Separabilità:** le visualizzazioni PCA e t-SNE di Run A appaiono più nitide e con cluster ben distinti, specialmente in t-SNE (Figura **??**), che presenta due nuvole compatte e minor overlap. Run B, pur mostrando due raggruppamenti, rivela sovrapposizioni maggiori e densità meno omogenee.

- **Compattezza dei Cluster:** i centroidi di Run A sono più distanti l'uno dall'altro in entrambe le proiezioni, indicando embedding più discriminativi, mentre in Run B la distanza inter-centroide risulta inferiore.

- **Implicazioni pratiche:** un modello fine-tuned fornisce embedding più adatti a separare categorie di risposta rilevanti per la sicurezza; l'embedding non fine-tuned, non progettato per tale compito, porta a raggruppamenti meno informativi.

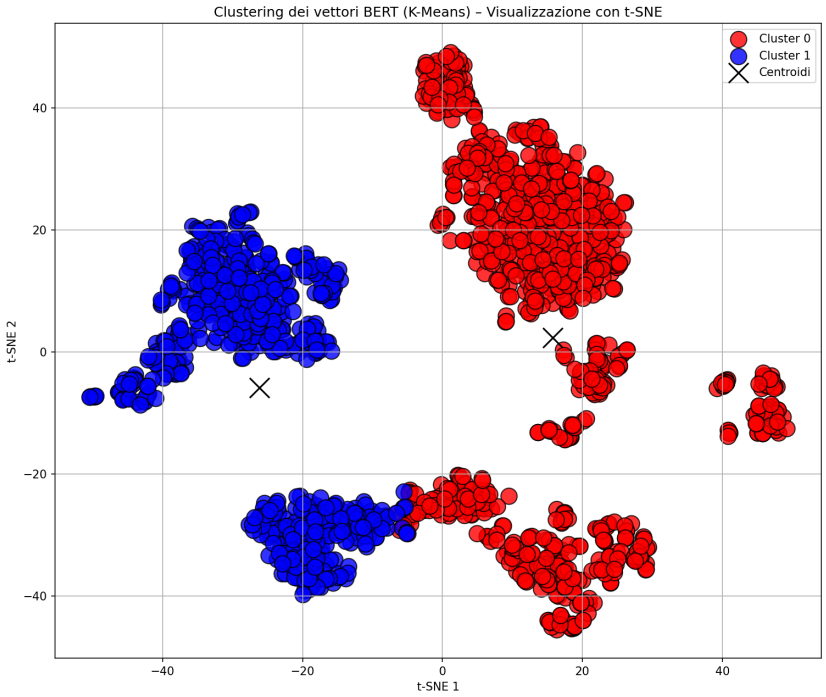

Figure 3: t-SNE dei cluster estratti dal BERT Fine-Tuned (Run A).

5 Conclusioni

Il fine-tuning di BERT sulle risposte jailbreak incrementa significativamente la qualità degli embedding per il task di distinzione non supervisionata, come evidenziato da distribuzioni più bilanciate e cluster più separati. Questi risultati suggeriscono l'opportunità di adottare modelli specificamente addestrati quando si desidera applicare tecniche di clustering per l'analisi di contenuti LLM in ambienti di produzione.

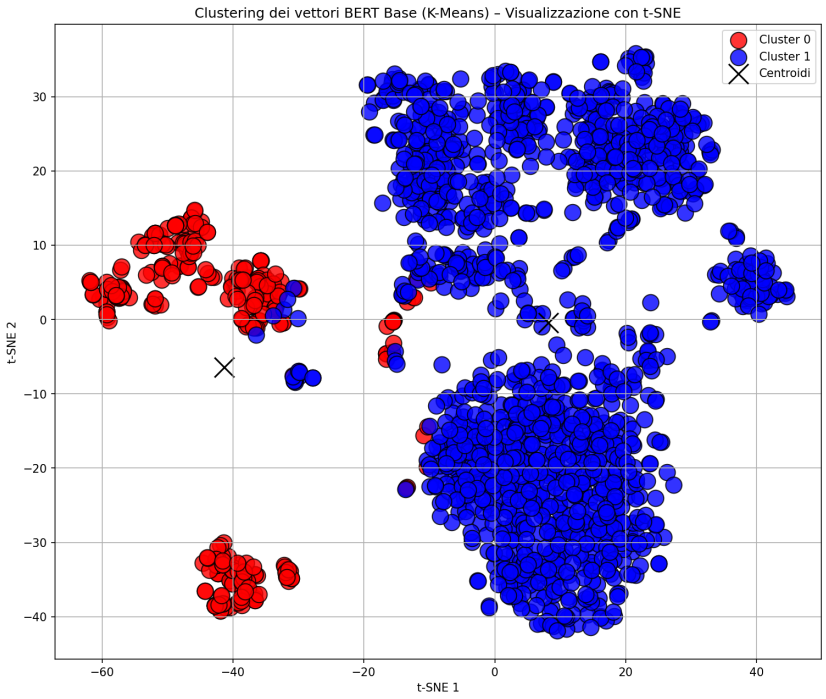

Figure 4: t-SNE dei cluster estratti dal BERT Base (Run B).