



Progetto Alstom

Gruppo 2 :

- Alesi Nicol
- Borino Marco
- De Pietri Ludovica
- Lapi Klima
- Papalia Michele
- Sansossio Matteo



Programma



Introduzione

Approccio al problema

Risoluzione proposta

- Script
- Tempi e costi

Possibili miglioramenti

- GPT 4/5 vs Ollama
- Database relazionale
- Interfaccia utente

Conclusioni



Introduzione alla challenge

**Automatizzare l'analisi dei guasti
avvenuti su un treno per capirne
le cause al fine di permettere al
cliente di gestire la situazione
presente e futura**



Working Flow

Input testuale e Input diagnostica

Input testuale, Input
diagnostica generico

Analisi input

Analisi dell'input testuale
con modelli di NLP, per
identificare parole chiave
legate ai guasti

Verifica file diagnostica

Verifica nel file di diagnostica
generale la presenza di
eventuali guasti, con focus
sulla colonna
precedentemente rilevata.

Caricamento diagnostica

Caricamento della
diagnostica specifica
dell'impianto coinvolto, così
da analizzare i
parametri specifici per
trovare la causa esatta

Risultato

Output strutturato in
linguaggio naturale
basato sui dati rilevati

Esempio: il modello
associa il prompt "WC
non funzionante", alla
colonna TOI1 (impianto
WC) del file di
diagnostica generico

Esempio: Se viene
trovato un guasto ("TOI1:WC
Fuori servizio"=1), il sistema
conferma il problema
e identifica l'impianto
coinvolto

Esempio: Il WC non funziona
perché il livello dell'acqua è
inferiore al 10 percento,
suggeriamo di controllare il
riempimento del serbatoio
TOI1 o verificare la pompa di
alimentazione.

Strategia tecnica

1

Analisi NLP della segnalazione

- Estrazione delle parole chiave dal plain text
- Mappatura delle parole chiave alle colonne della diagnostica generale

2

Verifica del guasto nel report generale

- Trova la colonna binaria corrispondente alla colonna individuata
- Se il valore è 1, conferma il guasto e passa al file dettagliato dell'impianto

3

Analisi del report dell'impianto

- Identifica la causa el guasto nei dati associati all'impianto

4

Verifica del guasto nel report di dettaglio

- Produc un output testuale chiaro e leggibile per spiegare il problema sfruttando gli LLM

Risoluzione proposta

Linguaggio : Python

Librerie principali:

Spacy

Elabora il linguaggio naturale (NLP), sfruttando modelli pre-addestrati (it_core_news_lg). Nel nostro caso analizza la similarità semantica.

Pandas

Analisi dati strutturati, necessaria per manipolare e leggere file xlsx o csv.

tkinter

Permette la creazione di interfacce grafiche utente :

- Finestre
- Bottoni
- Etichette

Ollama

Consente di eseguire modelli di intelligenza artificiale localmente, senza dover dipendere dalle API cloud di terze parti.

Script

analisiTesto.py

- Prende in input il file di testo
- Prende in input il file excell di diagnostica generico
- Identifica una connessione tra il testo inserito e la colonna d'errore
- Verifica il flag d'errore nella colonna
- Se c'è effettivamente un errore viene eseguito lo script analisiDettaglioRaffinata.py

analisiDettaglioRaffinata.py

- Analizza il file excell di dettaglio (ora è fisso dovrebbe poi diventare parametrico)
- Genera un report attraverso l'utilizzo del modello mistral (LLM) basandosi sulle indicazioni di un **prompt**
- Ricerca le colonne dei contatori e ne stampa il valore minimo e massimo
- Salva il report in un file .txt e lo mostra a schermo

analisiDettaglio.py

Ad oggi non è stata compresa nel working flow degli script, logicamente molto simile ad analisiDettaglioRaffinata, ma elabora un prompt più standard attraverso l'utilizzo di numpy. I due metodi potrebbero essere fusi insieme per ottenere un risultato migliore.

Script - Esempio I/O

Questo PC > Volume (D:) > Alstom > docs

Report e Contatori

Report di Stato degli Impianti

Data e Ora: [Data e ora tratta dal file]

Durante la giornata, si è verificato il seguente stato degli impianti:

1. Inizialmente, l'impianto WC era in servizio ed era libero. L'impianto di scarico dei rifiuti (sistema scarico tazza) non ha presentato problemi all'inizio.
2. Successivamente, si è verificata l'isolamento del WC, che è passato quindi in stato "fuori servizio". In contemporanea con questo evento, è avvenuto il reset dello stato del sistema scarico tazza.
3. Più avanti nella giornata, si è rilevato un'anomalia nel sistema scarico tazza per cui lo stato è passato a "occupato". Al contempo, la porta ha subito un'apparizione anomala nel sistema scarico tazze per cui lo stato è stato "reset".
4. Successivamente, si sono verificati molti avvicendamenti "e lo stato "in servizio" per il sistema scarico tazza e la porta. Tuttavia, non sono state rilevate anomalie nel sistema scarico tazze per cui lo stato è stato "reset".
5. Infine, si è verificata l'isolamento del WC che ha fatto tornare lo stato "fuori servizio". In contemporanea con questo evento, la porta di scarico dei rifiuti è tornata a essere libero.

Conclusioni: Al termine della giornata, si ha una situazione in esame. Il WC è stato in generale in uno stato "fuori servizio" e continuamente in uno stato anomalo, sebbene la porta abbia superato i problemi riscontrati.

TOI:Contatore Attivazioni Sciacquone -> Min: 6415.0, Max: 6485.0
TOI:Contatore Attivazioni Rubinetto -> Min: 3932.0, Max: 3972.0
TOI:Contatore Operazioni Lavandino -> Min: 4390.0, Max: 4408.0
TOI:Contatore Attivazioni Asciugami -> Min: 1166.0, Max: 1171.0
TOI:Contatore Operazioni Pompa -> Min: 6243.0, Max: 6272.0
TOI:Contatore Porta Chiusa e Bloccata -> Min: 3768.0, Max: 3810.0
TOI2:Contatore Attivazioni Sciacquone -> Min: 10603.0, Max: 10603.0
TOI2:Contatore Operazioni Lavandino -> Min: 8541.0, Max: 8543.0
TOI2:Contatore Operazioni Pompa -> Min: 5826.0, Max: 5932.0
TOI2:Contatore Porta Chiusa e Bloccata -> Min: 6127.0, Max: 6127.0

Inserisci il testo: .Treno A viaggiava in accoppiata con Treno B

Esegui Analisi

Analisi Dati WC Fuori Servizio

Inserisci il testo: Non mi interessa niente

Esegui Analisi

Errore

Nessuna corrispondenza trovata nel testo.

OK

OK

Torna al Programma

Tempi e costi

Ollama fornisce diversi modelli preaddestrati di LLM, verticalizzabili, ma con caratteristiche di base diverse.

Mistral

- Minimo 8gb RAM

Ambiente di test

- 16GB RAM
- i5-12500H
- NVIDIA GeForce RTX 4050 Laptop
- Windows 11
- Ollama 0.6.2

Caso peggiore :

16.896 righe non nulle (il doppio dell'attuale file di dettaglio)



15 token per frase

$$\text{Tempo} = \frac{253.440}{38,936} \sim 6.509,143 \text{ s} \sim 108,485 \text{ m}$$

↑ token
↓ token/s

[Torna al Programma](#)

```
System: Intel(R) Core(TM) i5-12500H CPU @ 12.00GHz, 15.65 GB
CPU: Intel(R) Core(TM) i5-12500H CPU @ 12.00GHz, 15.65 GB
GPU: NVIDIA GeForce RTX 4050 Laptop GPU Intel(R) 12.00GHz, 8.00 GB
OS: Microsoft Windows 11 Home

Version: 0.6.2

Running custom benchmark from models_file_path: C:\Users\HP\Downloads\Deepserk\Deepserk\Deepserk\Deepserk\models
Using sendinfo for custom benchmark
Models file path: C:\Users\HP\AppData\Local\Programs\Ollama\models
Fetching and pulling the following LLM models
mistral

Running custom-model
model_name = mistral
prompt = Summarize the key differences between classical and operant conditioning.
rate: 38.99 tokens/s
prompt = Translate the following English paragraph into Chinese
rate: 39.27 tokens/s
prompt = What are the main causes of the American Civil War?
rate: 39.11 tokens/s
prompt = How does photosynthesis contribute to the carbon cycle?
rate: 39.04 tokens/s
prompt = Develop a python function that solves the following equation
rate: 38.27 tokens/s
rate: 38.936 tokens/s
```

Possibili miglioramenti

Ollama vs ChatGPT 4/5

Confronto tra il metodo impiegato nel codice proposto ed I.A. maggiormente addestrate.

Database relazionale

Possibili miglioramenti applicati al database, per avere una collezione e gestione di dati più efficiente.

Interfaccia utente

Miglioramento proposto ai fini di ottenere un'interfaccia più user friendly ed esteticamente migliore.

Confronto tra Ollama e GPT-4/5 per l'implementazione aziendale di un LLM



OLLAMA

CHATGPT4/5

Esecuzione

Locale: il modello gira direttamente su PC o server aziendale, senza necessità di connessione Internet.

Cloud: il modello è accessibile via Internet, su server OpenAI, attraverso API o interfacce web.

Privacy

Massimo controllo: i dati restano sempre all'interno dell'azienda, fondamentale per la sicurezza industriale.

I dati viaggiano verso server esterni, anche se cifrati. **Meno controllo sulla riservatezza.**

Costi

Gratuito: non richiede costi di utilizzo, salvo quelli hardware (RAM, GPU). Ideale per soluzioni interne.

A pagamento: il costo è calcolato in base ai token elaborati. Può diventare oneroso su grandi volumi.

Velocità di risposta

Molto veloce in locale, senza latenza di rete. Prestazioni dipendono dall'hardware disponibile.

Molto veloce, ma **dipende da rete e server esterni**. Può subire rallentamenti in caso di alto traffico.

Addestrabilità

Alta flessibilità: è possibile effettuare fine-tuning o adattare il modello ai dati specifici dell'azienda.

Limitato: non consente fine-tuning diretto. Possibili personalizzazioni solo tramite prompt o GPT personalizzati.

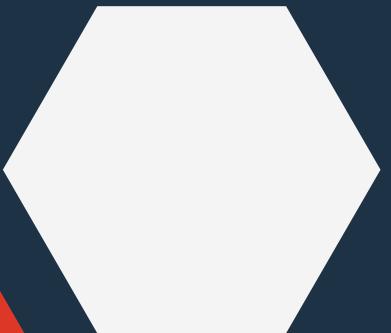
Potenza del modello

Modelli leggeri (es. Mistral, LLaMA): ottimi per task specifici, **meno potenti per ragionamenti complessi**.

Modello estremamente potente: eccelle in compiti complessi, ragionamenti articolati e linguaggio naturale.

Database relazionale

Linguaggio SQL e Python



Analisi dei passaggi:

Tipicamente i file vengono gestiti seguendo i passaggi di:

- Input testuale e Input diagnostica
- Analisi input
- Verifica file diagnostica
- Caricamento diagnostica
- Risultato

Creazione di filtri appositi in SQL

Che cos'è un filtro in un Database?

Un filtro programmato in SQL è un tool appositamente creato dai gestori e ideatori del Database per filtrare, raggruppare e gestire i dati contenuti in esso, in qualsiasi loro forma o dimensione. Ad oggi i filtri in ambito della gestione dati sono ancora una delle risorse più preziose che i programmatore di strutture logiche relazionali utilizzano.

Database relazionale

Linguaggio SQL e Python

Possibili impieghi:

In un'azienda in cui vi sono ingenti quantità di dati giornalieri come Alstom questi tool permetterebbero di snellire quelle che sono alcune procedure nell'analisi e gestione dei dati. Alcune pratiche applicazioni sono:

Creazione di dataset “puliti”

Gestione avanzata del Database e dei dati;

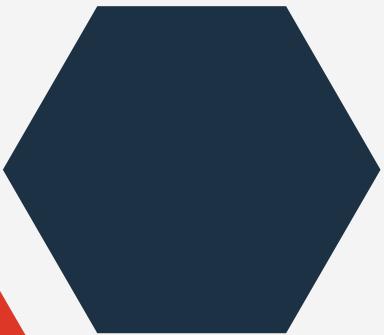
Efficientamento nella ricerca dei dati;

Separazione delle tipologie di dati;

Questi quattro applicazioni efficienterebbero in modo consistente quelli che sono i tempi di output e di addestramento anche dell'Intelligenza Artificiale prevista per il progetto (previa integrazione dell'Algoritmo/Intelligenza con il DB con apposita libreria come psycopg 2 nel caso d'uso di SQL).

Database relazionale

Linguaggio SQL e Python



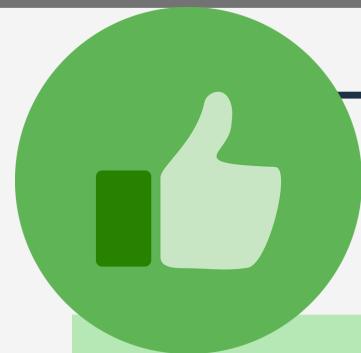
Possibili impieghi

In un'azienda in cui vi sono ingenti quantità di dati giornalieri come Alstom questi tool permetterebbero di snellire quelle che sono alcune procedure nell'analisi e gestione dei dati. Alcune pratiche applicazioni sono:

- Creazione di dataset “puliti”
- Gestione avanzata del Database e dei dati;
- Efficientamento nella ricerca dei dati;
- Separazione delle tipologie di dati;

Questi quattro applicazioni efficienterebbero in modo consistente quelli che sono i tempi di output e di addestramento anche dell’Intelligenza Artificiale prevista per il progetto (previa integrazione dell’Algoritmo/Intelligenza con il DB con apposita libreria come psycopg 2 nel caso d’uso di SQL).

Pro e contro per l'utilizzo dei filtri nelle strutture logiche



PRO

Migliora le performance: Si riducono i dati a quelli realmente necessari → meno memoria e meno tempo di elaborazione

Maggiore chiarezza del dato: Filtri ben scritti rendono il risultato più leggibile, pulito e coerente con l'obiettivo dell'analisi.

Automazione facile: Una volta definiti i filtri, questi possono essere riutilizzati o schedulati senza intervento manuale.

CONTRO

Rischio di perdita di informazioni utili: Filtri troppo restrittivi → potresti escludere dati importanti senza accorgertene.

Complessità nella manutenzione: Se i filtri sono molti o complicati, aggiornare o modificare il codice può diventare difficile e rischioso.

Errori logici o sintattici: Un errore nei filtri (es. condizioni sbagliate) può portare a risultati completamente errati senza segnali evidenti.



Esempi di pratici di filtri

```
SELECT *
FROM clienti
WHERE città = 'Milano';
```

E' importante sottolineare che i filtri programmati in SQL, possono essere arricchiti da una serie di correlatori logici che li possono rendere ancora più specifici, tra i più importanti si sottolineano:

- And
- Or
- If
- Else

Dati esemplificativi

```
clienti = [
    {"id": 1, "nome": "Luca", "città": "Milano", "età": 30},
    {"id": 2, "nome": "Anna", "città": "Roma", "età": 25},
    {"id": 3, "nome": "Marco", "città": "Milano", "età": 40},
]
```

```
milano_clienti = [c for c in clienti if c["città"] == "Milano"]

print(milano_clienti)
```



UPLOAD DEL FILE EXCEL

Carica il file Excel che contiene l'elenco dei guasti del treno. Il sistema analizzerà i dati per aiutarti a trovare le cause e le soluzioni

BARRA DI RICERCA

Scrivi una parola chiave, un codice guasto o una descrizione del problema per ottenere le possibili cause o soluzioni dai dati caricati



UPLOAD DEL FILE EXCEL

Carica il file Excel che contiene l'elenco dei guasti del treno. Il sistema analizzerà i dati per aiutarti a trovare le cause e le soluzioni.

BARRA DI RICERCA

Scrivi una parola chiave, un codice guasto o una descrizione del problema per ottenere le possibili cause o soluzioni dai dati caricati





UPLOAD DEL FILE EXCEL

Carica il file Excel che contiene l'elenco dei guasti del treno. Il sistema analizzerà i dati per aiutarti a trovare le cause e le soluzioni.

BARRA DI RICERCA

Scrivi una parola chiave, un codice guasto o una descrizione del problema per ottenere le possibili cause o soluzioni dai dati caricati

Suggerimento: utilizza i link per inserire collegamenti alle diverse risorse aziendali.

Come: seleziona il testo, clicca sul simbolo del link nella barra degli strumenti e incolla il collegamento al documento o ai siti web desiderati.

Grazie per l'attenzione!