Università Politecnica delle Marche

Facoltà di Ingegneria

Dipartimento di Ingegneria dell'Informazione

Corso di Laurea Magistrale in Ingegneria Informatica e dell'Automazione

Applicazione tecniche di NLP su un dataset contenente recensioni di articoli tecnologici

Docenti
Prof. Ursino Domenico
Dott. Buratti Christopher

Componenti del gruppo
Dott. Tempera Fabio
Dott. Marcianesi Luca
Dott. Vianello Gabriele

ANNO ACCADEMICO 2024-2025

Indice

Elenco delle figure

Elenco delle tabelle

1 Introduzione al Natural Language Processing (NLP)

Il Natural Language Processing (NLP) è un campo dell'intelligenza artificiale che si occupa dell'interazione tra computer e linguaggio umano. L'obiettivo primario di questa disciplina è permettere ai sistemi informatici di comprendere, analizzare, generare e rispondere al linguaggio naturale in modo significativo e utile. Nel contesto del progetto Faboulous-Interpretr, queste tecniche vengono applicate a due compiti distinti ma complementari: la sintesi documentale strutturata e l'analisi della salute mentale attraverso la classificazione di testi.

Le componenti fondamentali del NLP che interessano il nostro lavoro sono le seguenti:

1. **Comprensione del Linguaggio Naturale (NLU - Natural Language Understanding)**:
    - si concentra sul decifrare il significato del testo scritto o parlato;
    - include attività come l'analisi semantica e la classificazione del testo;
    - nel nostro progetto, viene utilizzata per identificare stati emotivi e psicologici all'interno di messaggi o post, classificandoli in categorie specifiche come depressione, ansia o stress.
2. **Generazione del Linguaggio Naturale (NLG - Natural Language Generation)**:
    - consente ai computer di creare nuovi testi che siano comprensibili e coerenti per un lettore umano;
    - esempi classici sono la traduzione automatica e la creazione di riassunti;
    - il nostro sistema sfrutta modelli seq2seq come IT5 per generare sintesi accurate di documenti tecnici e specifiche API.

3. **Elaborazione del Linguaggio Naturale (NLP - core processing)**:
    - riguarda il trattamento preliminare dei dati per renderli digeribili dagli algoritmi;
    - include compiti come la pulizia del testo, la segmentazione (chunking) e la tokenizzazione;
    - la pipeline del progetto utilizza strategie avanzate come il chunking ricorsivo per preservare la coerenza semantica dei documenti lunghi prima dell'analisi.

Per raggiungere questi obiettivi utilizziamo diverse metodologie, spaziando dalle tecniche classiche a quelle più recenti basate sui Transformer:

- **Metodi basati su Embeddings**: trasformano le parole in vettori numerici densi che catturano relazioni semantiche. Abbiamo esplorato l'uso di GloVe (Global Vectors for Word Representation) addestrati su corpus differenti per confrontare le prestazioni su testi formali e informali.
- **Reti Neurali Ricorrenti (RNN)**: architetture progettate per trattare dati sequenziali. Nel progetto è stata implementata una rete LSTM (Long Short-Term Memory) bidirezionale come baseline per valutare i miglioramenti apportati dai modelli più moderni.
- **Transformer e Transfer Learning**: l'attuale stato dell'arte del NLP. Utilizziamo modelli pre-addestrati come XLM-RoBERTa e IT5, dove il primo è stato adattato ai nostri scopi specifici tramite tecniche di fine-tuning efficiente come LoRA (Low-Rank Adaptation), che permettono di ottenere alte prestazioni con un costo computazionale contenuto.

1.1 Tecnologie e Framework Utilizzati

Per lo sviluppo della piattaforma Faboulous-Interpretr, abbiamo selezionato un insieme di tecnologie che garantiscono modularità, efficienza e facilità di utilizzo:

- **Python 3.13+**: linguaggio di riferimento per il progetto, essenziale per l'accesso al vasto ecosistema di librerie per la data science e il machine learning.
- **PyTorch**: framework di deep learning utilizzato come backend computazionale per l'addestramento e l'inferenza dei modelli neurali.

- **Hugging Face Transformers**: libreria centrale per l'accesso ai modelli Transformer pre-addestrati (come IT5 e XLM-RoBERTa) e agli strumenti di tokenizzazione.
- **PEFT (Parameter-Efficient Fine-Tuning)**: libreria fondamentale per implementare la tecnica LoRA, permettendo di addestrare modelli di grandi dimensioni su hardware consumer.
- **Streamlit**: framework per la creazione rapida dell'interfaccia utente web, che permette di interagire con i modelli di sintesi e classificazione in tempo reale.
- **Trafilatura e PyMuPDF**: strumenti specifici per l'ingestione dei dati, utilizzati rispettivamente per l'estrazione pulita di testo da pagine web e da documenti PDF.

1.2 Applicazioni nel Contesto del Progetto

Le tecnologie NLP implementate trovano applicazione pratica in due aree principali all'interno della nostra piattaforma:

- **Sintesi Documentale Intelligente**: il sistema è in grado di processare documenti tecnici (e non) eterogenei, come specifiche OpenAPI, pagine web e file PDF. Attraverso una strategia di Map-Reduce, il testo viene suddiviso, analizzato e ricomposto per fornire all'utente un riassunto coerente che mantiene i punti salienti del contenuto originale.
- **Monitoraggio della Salute Mentale**: mediante l'analisi di testi liberi o caricamento massivo di file csv contenenti messaggi, il sistema identifica segnali linguistici associati a diversi stati psicologici. Questa applicazione dimostra come il NLP possa essere utilizzato per supportare l'analisi preliminare di grandi volumi di dati testuali in ambito socio-psicologico, filtrando i contenuti in base alla gravità o alla tipologia di emozione espressa.

2 Introduzione al progetto

Il progetto Faboulous-Interpretr nasce come una piattaforma NLP altamente modulare, progettata per affrontare due delle sfide più attuali nell'elaborazione del linguaggio naturale: la sintesi documentale di testi tecnici complessi e l'analisi psicologica di contenuti testuali liberi. Sviluppato nell'ambito di un corso universitario di Data Science, il sistema mira a fornire uno strumento "production-ready" in grado di operare con efficienza anche su hardware consumer, grazie all'impiego modelli leggeri e fine tuning LoRA.

A differenza dei sistemi generalisti, questa piattaforma è stata costruita con un approccio modulare che separa nettamente la fase di ingestione dei dati, il motore di elaborazione neurale e l'interfaccia utente. L'obiettivo è trasformare dati non strutturati, provenienti da fonti eterogenee come PDF o URL web, in informazioni strutturate e azionabili, sia che si tratti di un riassunto tecnico o di una valutazione sullo stato di salute mentale di un autore.

2.1 Architettura del Sistema

Il sistema è strutturato secondo un'architettura logica a tre livelli, che guida il flusso dei dati dall'input all'output finale. L'intera applicazione è orchestrata tramite un'interfaccia web interattiva sviluppata in `Streamlit`, che funge da punto di accesso per l'utente finale.

La dashboard principale, visibile in Figura 2.1, offre un accesso centralizzato ai moduli di "Technical Summarization" e "Review Sentiment Analysis", fornendo inoltre un feedback immediato sull'hardware in uso (nel caso specifico, accelerazione CUDA attiva).

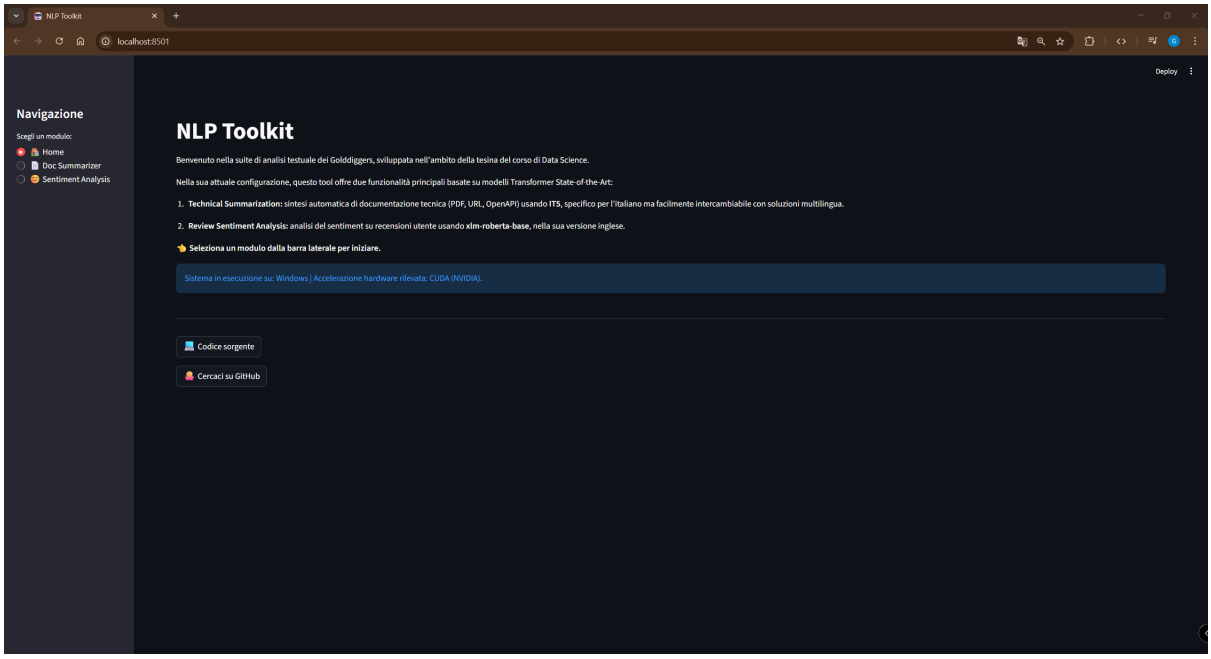

Figura 2.1: Home page dell'applicazione NLP Toolkit con rilevamento hardware attivo.

Le componenti principali dell'architettura sono le seguenti:

- **Input Layer (Data Ingestion)**: modulo responsabile dell'acquisizione e della normalizzazione del testo. Include adattatori specifici per diverse fonti:
  - loader per file PDF basato su PyMuPDF per l'estrazione fedele del contenuto;
  - scraper web basato su Trafilatura per ottenere testo pulito da URL, eliminando elementi di navigazione e pubblicità;
  - parser per specifiche OpenAPI (JSON/YAML) che converte descrizioni tecniche in linguaggio naturale discorsivo.
- **Processing Layer (NLP Core Engine)**: il cuore computazionale del sistema, diviso in due pipeline distinte:
  - pipeline di summarization basata sul modello IT5 (T5 per l'italiano), che utilizza una strategia Map-Reduce per gestire documenti che superano la lunghezza massima gestibile dal modello;
  - pipeline di classificazione mental health basata su XLM-RoBERTa, potenziata con adapter LoRA per identificare stati emotivi con elevata accuratezza e ridotto consumo di memoria.
- **Output Layer (User Interface)**: dashboard realizzata con Streamlit che permette di visualizzare i riassunti generati, i grafici relativi all'analisi del sentiment e le metriche di confidenza del modello.

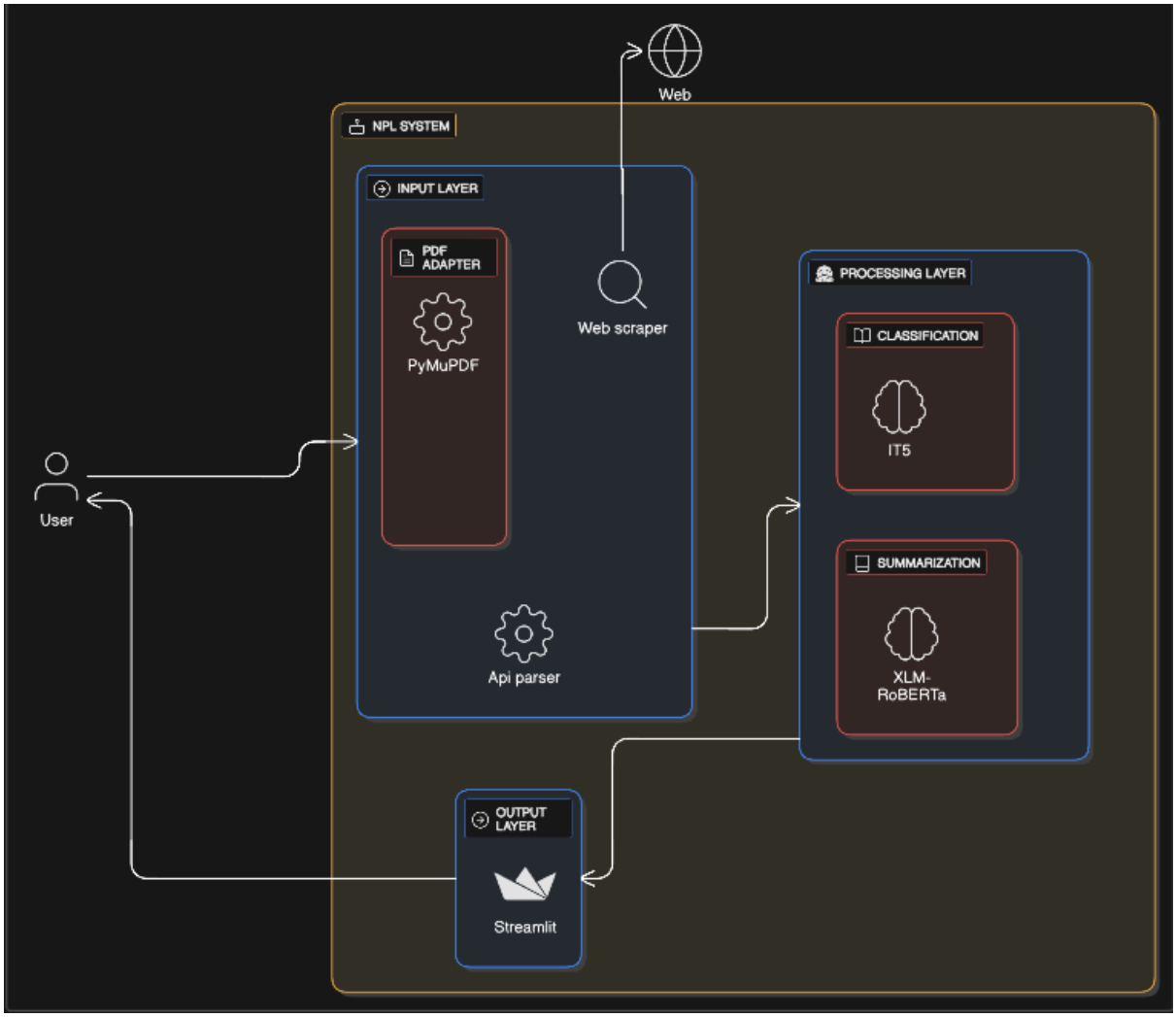

Figura 2.2: Architettura dell'applicazione

2.2 NLP: Dataset utilizzato

Per la componente di analisi della salute mentale, il progetto si avvale di un dataset specifico contenente messaggi e post estratti da piattaforme social e forum di supporto. Il dataset originale, denominato `mental.csv`, è costituito da due colonne principali che associano un testo libero a un'etichetta diagnostica o emotiva.

I dati sono stati trattati per permettere l'addestramento di modelli supervisionati in un contesto di classificazione multiclasse. Di seguito è riportata una descrizione delle feature presenti nel dataset grezzo:

Feature	Descrizione
statement	Il contenuto testuale del messaggio o del post scritto dall'utente, che può variare da poche parole a lunghi sfoghi emotivi.
status	L'etichetta originale assegnata al testo, che indica lo stato psicologico o la diagnosi associata (es. Anxiety, Depression, Normal).

Tabella 2.1: Descrizione delle feature del dataset di analisi mental health

2.2.1 Analisi Descrittiva e Bilanciamento del Dataset

Prima dell'addestramento, è stata condotta un'analisi esplorativa (EDA) sul dataset `mental.csv` per comprendere la distribuzione delle categorie. Come evidenziato in Figura 2.3, il dataset presentava un forte sbilanciamento.

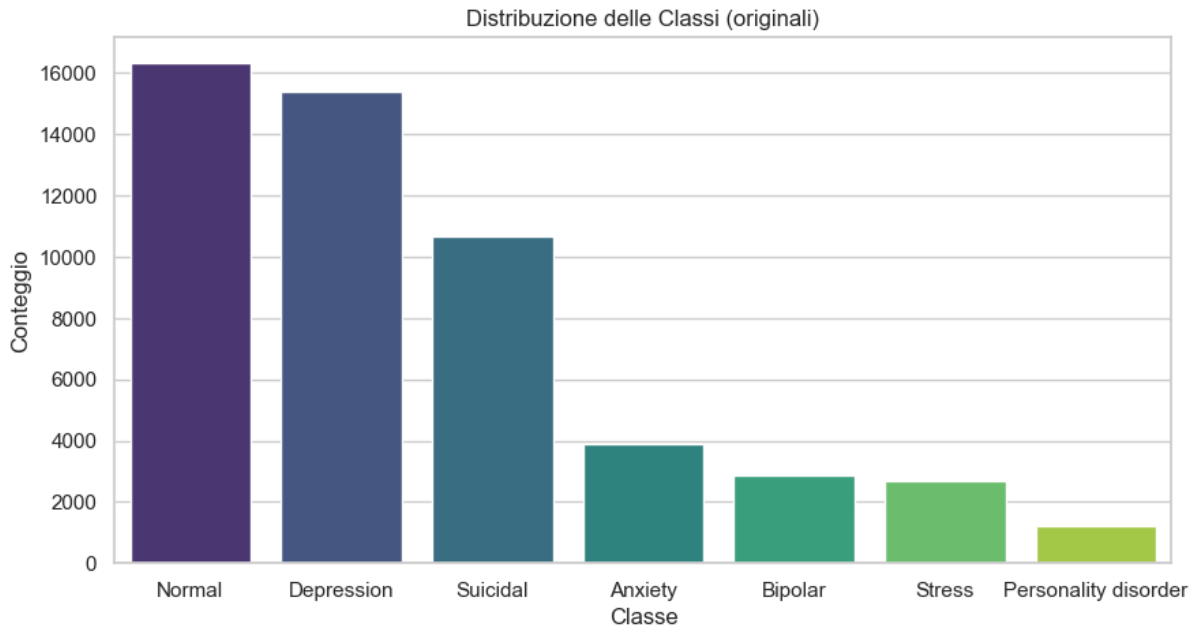

Figura 2.3: Distribuzione originale delle classi: si nota una forte prevalenza della classe Normal.

La categoria "Normal" risultava essere sovra-rappresentata (oltre 16.000 campioni), mentre classi critiche come "Personality disorder" contavano poche migliaia di esempi. Un tale sbilanciamento rischiava di indurre il modello a predire quasi sempre la classe maggioritaria, ignorando i segnali di disagio meno frequenti.

Per mitigare questo problema, è stata applicata una strategia di undersampling controllato unita all'aggregazione delle classi simili ("Anxiety" e "Stress" nella macro-categoria *Light*; "Bipolar", "Suicidal" e "Personality Disorder" nella macro-categoria *Serious*).

Il risultato è visibile in Figura 2.4: un dataset equilibrato con circa 8000 campioni per le classi principali, che permette al modello di apprendere caratteristiche distintive per ogni stato emotivo senza bias verso la classe dominante.

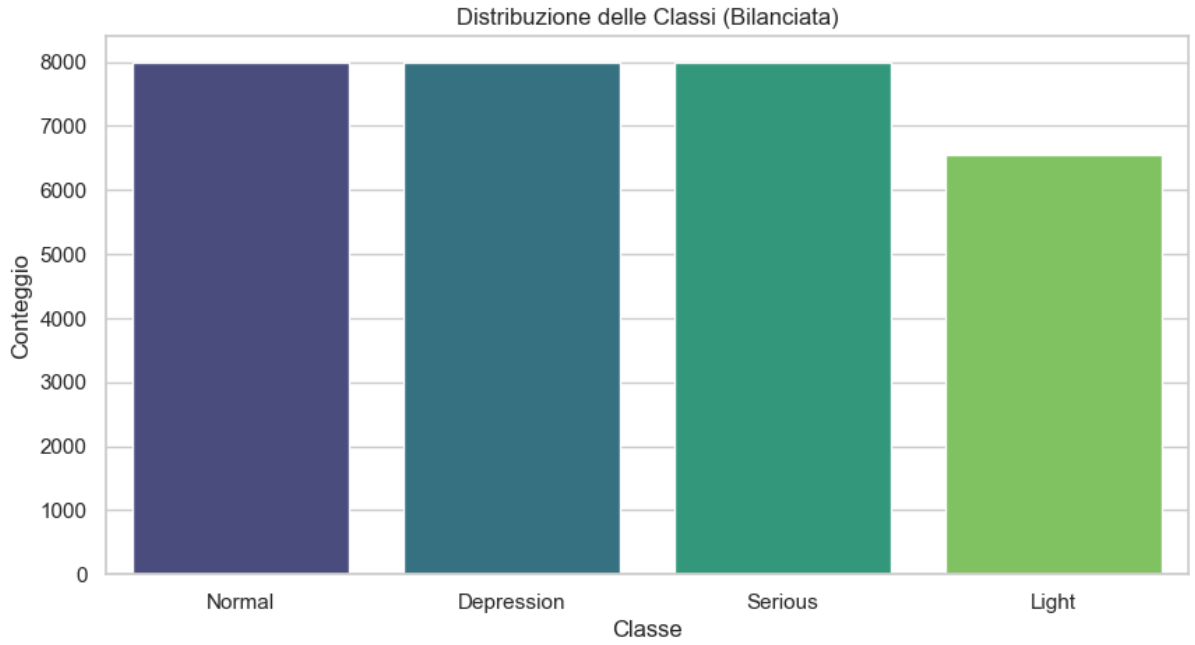

Figura 2.4: Distribuzione delle classi dopo il bilanciamento e l'aggregazione.

2.3 Pre-processing e Pipeline di Elaborazione

La qualità dell'output dei modelli NLP dipende strettamente dalla fase di preprocessing. In Faboulous-Interpretr, sono state implementate strategie differenziate per i due task principali.

2.3.1 Chunking Semantico Ricorsivo

Per la sintesi documentale, la sfida principale è gestire testi che superano la finestra di contesto del modello, per il modello utilizzato pari a 512 token (estremamente piccola se paragonata con quella di moderni LLM o SLM). A tal fine è stata sviluppata la classe `RecursiveTokenChunker`, che implementa un algoritmo di segmentazione gerarchica:

1. il sistema tenta di dividere il testo in corrispondenza dei doppi a capo (fine paragrafo);
2. se il segmento è ancora troppo lungo, prova a dividere sui singoli a capo;
3. successivamente cerca i confini delle frasi (punto seguito da spazio);

4. solo in ultima istanza ricorre alla divisione sugli spazi tra le parole.

Questo approccio garantisce che i ”chunk” (frammenti di testo) passati al modello mantengano un senso compiuto, evitando di troncare frasi a metà e migliorando la coerenza del riassunto finale.

3 Applicazione tecniche di NLP

3.1 Analisi visuale tramite WordCloud

Prima di procedere con l'addestramento dei modelli neurali, è stata effettuata un'analisi qualitativa del dataset attraverso l'uso di WordCloud. Questa tecnica di visualizzazione permette di rappresentare graficamente la frequenza dei termini all'interno di un corpus testuale, dove la dimensione di ogni parola è direttamente proporzionale al numero di volte che essa compare nei documenti.

Nel contesto del progetto Faboulous-Interpretr, l'uso delle WordCloud si è rivelato fondamentale per esaminare il vocabolario specifico associato alle diverse classi di salute mentale (Normal, Depression, Anxiety, ecc.) presenti nel training set.

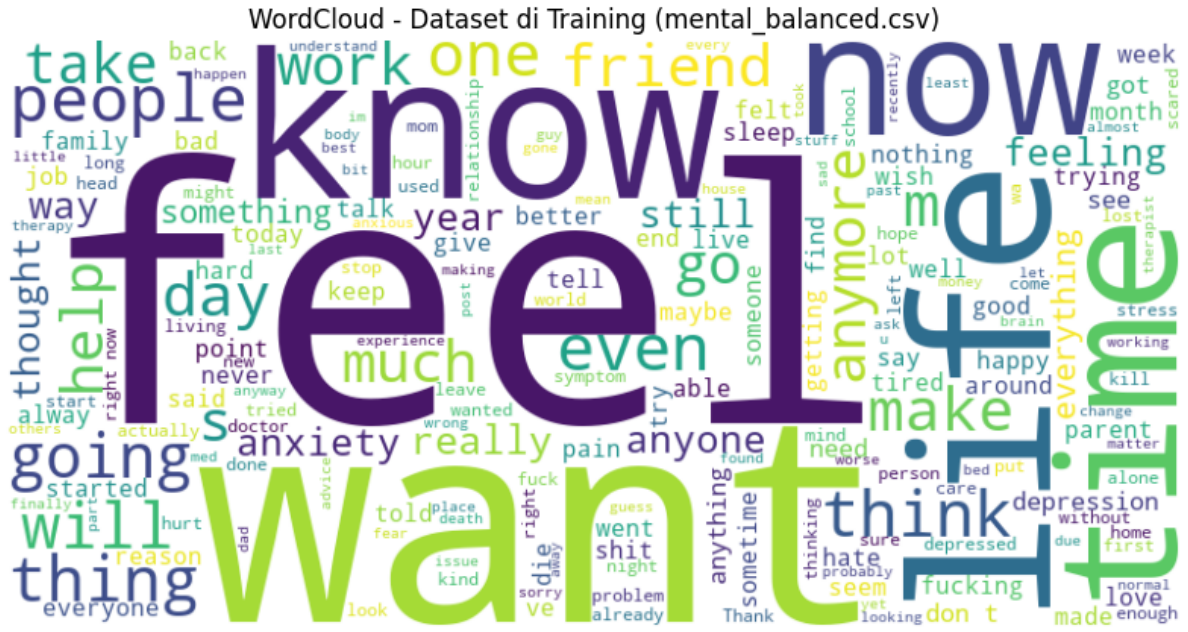

Figura 3.1: WordCloud dei termini più frequenti nel dataset di training.

L'osservazione della Figura 3.1 e delle visualizzazioni per singola classe ha evidenziato differenze lessicali significative. Mentre i testi etichettati co-

me ”Normal” tendono a contenere un vocabolario generico e variegato, le classi critiche mostrano una ricorrenza marcata di termini specifici legati allo stato emotivo e psicologico dell’autore.

3.2 Approcci Basati su Machine Learning Classico e RNN

Prima di implementare le architetture basate su Transformer, è stata condotta una fase sperimentale incrementale, partendo da modelli statistici fino ad arrivare alle reti ricorrenti, al fine di stabilire una baseline di performance solida.

3.2.1 Baseline: Naive Bayes

Il primo approccio tentato è stato l’utilizzo di un classificatore Naive Bayes (MultinomialNB), un algoritmo probabilistico basato sul teorema di Bayes. Per la rappresentazione del testo, abbiamo utilizzato un vettorizzatore TF-IDF (*Term Frequency-Inverse Document Frequency*) limitato alle 5000 feature più rilevanti e con rimozione delle stop-words inglesi.

Sebbene questo modello sia computazionalmente molto leggero, ha mostrato i limiti tipici degli approcci ”Bag-of-Words”: l’incapacità di catturare il contesto e l’ordine delle parole. Con un’accuratezza di circa il 58%, ha funto da ”baseline” minima da superare con i modelli successivi.

3.2.2 Reti Neurali Ricorrenti (LSTM) ed Embeddings

In primo luogo, è stata implementata una rete LSTM (Long Short-Term Memory) bidirezionale. A differenza del Naive Bayes, le LSTM gestiscono sequenze di dati mantenendo una memoria degli input passati. Come embeddings è stato scelto GloVe Twitter, data la natura del dataset e la sua migliore capacità nel gestire slang inglese ed emoticon (cioè il dominio del dataset di riferimento).

3.2.3 Modello Transformer

E’ stato implementato il codice necessario per eseguire il download e l’ottimizzazione del modello XLM-RoBERTa, modello transformer bidirezionale molto potente per il task di classificazione. Su di esso è stato operato un fine tuning con la tecnica LoRA (Low-Rank Adaptation) estremamente efficace ed efficiente su GPU consumer-grade. Nello specifico:

- attraverso la LoRA è stato possibile addestrare in modo mirato solo lo 0.4% dei parametri totali del modello originale.

- sono state sostituite le teste di classificazione per il numero di categorie del dataset, 4 dopo il bilanciamento.
- il fine tuning ha richiesto 20 minuti per 5 epoche, su NVIDIA RTX 3060 TI (8GB).

3.3 Summarization con Strategia Map-Reduce

Per il compito di sintesi documentale, il sistema utilizza il modello IT5 (una variante italiana del T5 di Google). Questo modello opera secondo un paradigma ”text-to-text”, dove sia l’input che l’output sono stringhe di testo.

La sfida principale nell’utilizzo di questi modelli è il limite della finestra di contesto. Per riassumere documenti lunghi, abbiamo implementato una strategia Map-Reduce:

1. **Fase di Map**: il documento originale viene suddiviso in segmenti semanticamente coerenti. Ogni segmento viene passato al modello IT5 che genera un micro-riassunto.
2. **Fase di Reduce**: i riassunti parziali vengono concatenati e strutturati mantenendo i riferimenti alle sezioni originali.

Un esempio concreto del risultato di questa strategia è mostrato in Figura 3.2, dove il documento sul ”Generative AI” è stato riassunto mantenendo una chiara divisione in sezioni logiche.

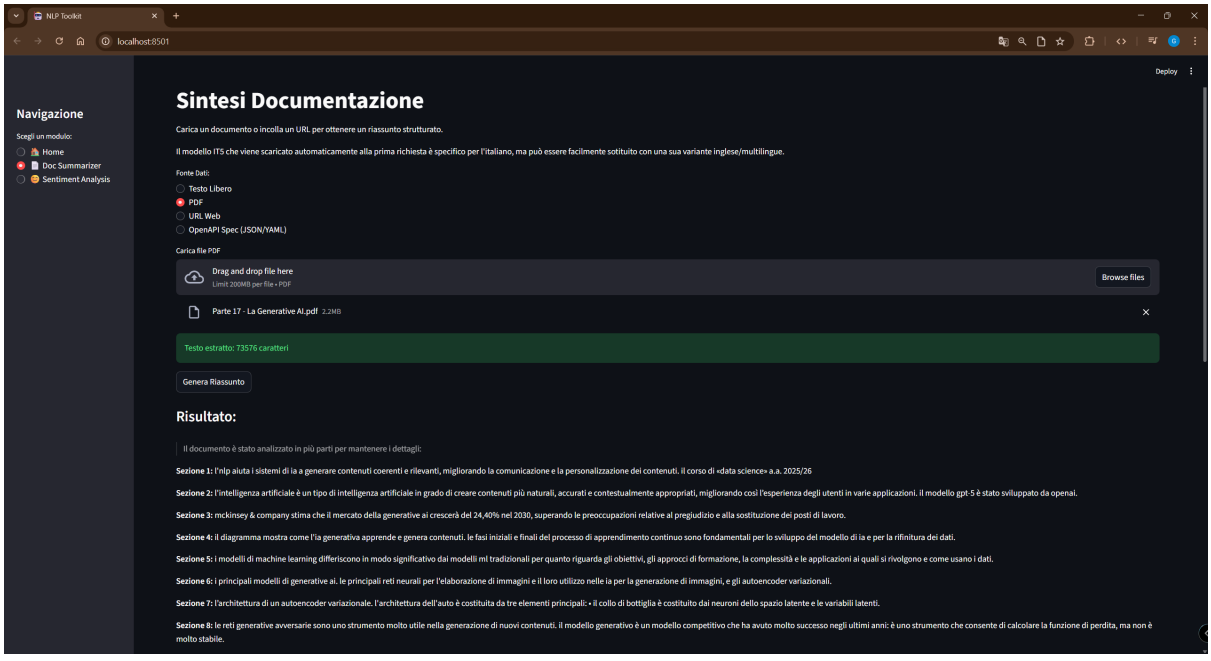

Figura 3.2: Output del modulo di sintesi: il testo è strutturato per sezioni numerate.

3.4 Interfaccia Utente e Deployment

L'accessibilità degli algoritmi è garantita da un'applicazione web sviluppata con Streamlit. Di seguito analizziamo nel dettaglio i flussi di lavoro implementati.

3.4.1 Modulo di Sintesi Documentale

Il modulo "Doc Summarizer" offre un'interfaccia flessibile per l'ingestione dei dati. L'utente può selezionare la fonte tra Testo Libero, PDF, URL Web o Specifiche OpenAPI, come mostrato in Figura 3.3.

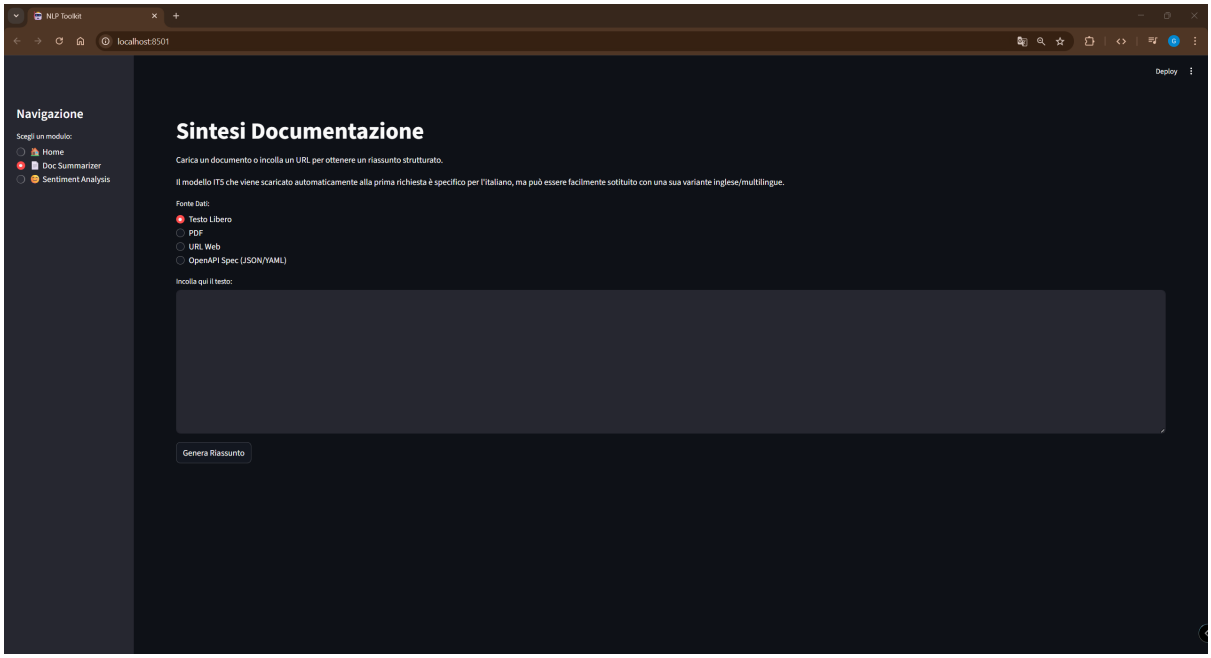

Figura 3.3: Interfaccia del modulo di sintesi con selezione della fonte dati.

Nel caso di caricamento di un file PDF (Figura 3.4), il sistema estrae automaticamente il testo e mostra il conteggio dei caratteri, fornendo un feedback immediato sull'avvenuta lettura del documento.

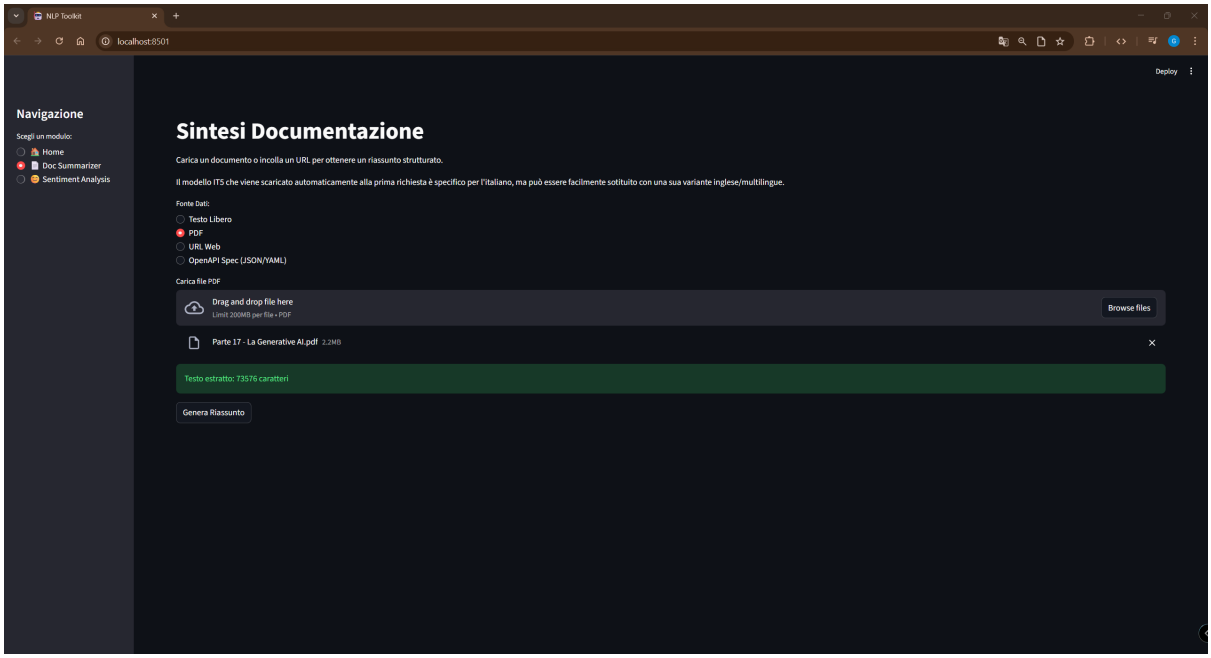

Figura 3.4: Upload di un file PDF e conferma dell'estrazione del testo.

Una volta avviata la generazione, il sistema mostra lo stato di avanzamento delle operazioni di caricamento del modello e inferenza (Figura 3.5), garantendo una buona user experience anche durante elaborazioni computazionalmente onerose.

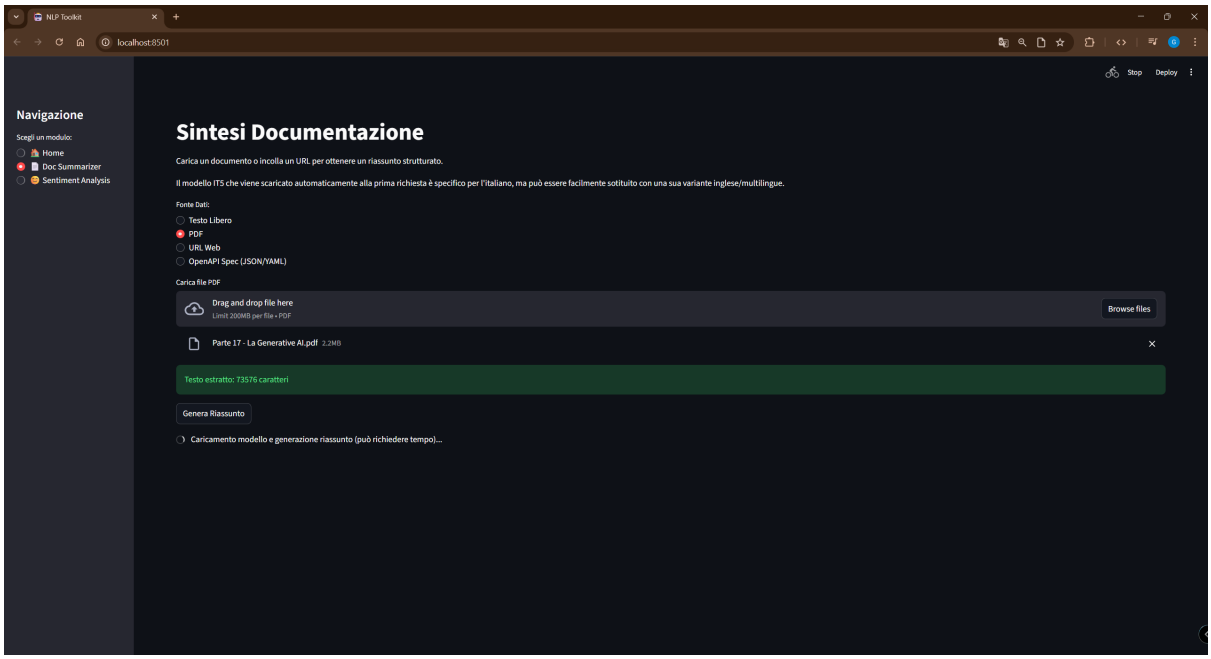

Figura 3.5: Indicatore di elaborazione durante la generazione del riassunto.

3.4.2 Modulo di Analisi Salute Mentale

Di seguito viene mostrata l'interfaccia principale della sezione di analisi della salute mentale, nella quale è possibile sia caricare una singola frase, che inserire un file in formato csv contenente una varietà di elementi da analizzare.

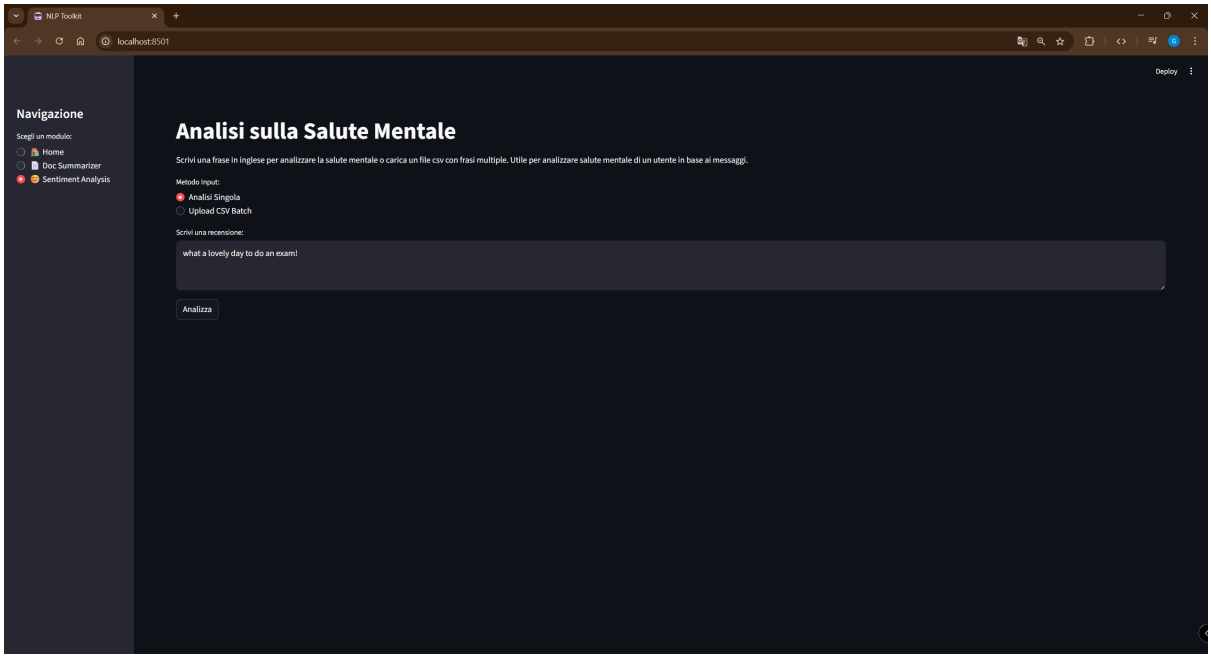

Figura 3.6: Interfaccia principale della sezione analisi salute mentale.

La sezione "Sentiment Analysis" permette come detto due modalità di utilizzo: puntuale e batch. Nella modalità singola, l'utente può inserire una frase in linguaggio naturale.

Il modello è in grado di distinguere sfumature sottili. Ad esempio, una frase positiva come "what a lovely day to do an exam!" viene correttamente classificata come **NORMAL** con altissima confidenza (0.99), come visibile in Figura 3.7.

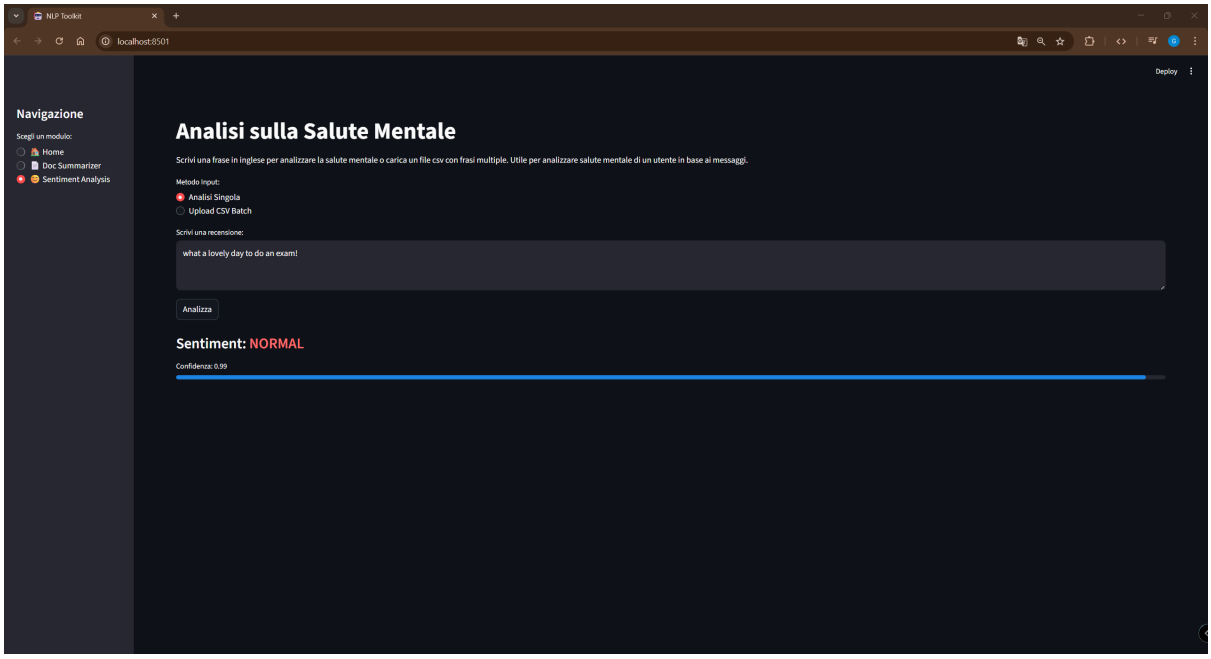

Figura 3.7: Classificazione corretta di una frase neutra/positiva (Classe: NORMAL).

Al contrario, frasi che esprimono ansia da prestazione (”i’m feeling stressed for this exam”) vengono identificate come **LIGHT** (Figura 3.8), categoria che raggruppa stati di ansia e stress moderato.

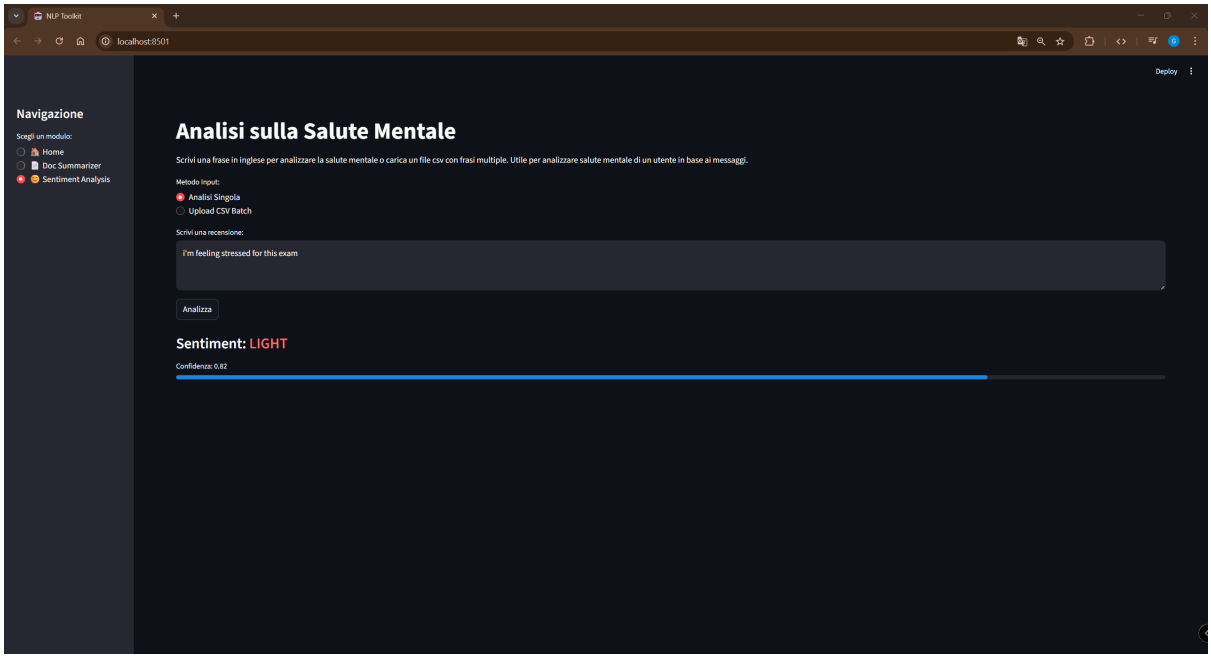

Figura 3.8: Rilevamento di ansia/stress (Classe: LIGHT).

Infine, espressioni che denotano un disagio più profondo o isolamento (”i’m lonely but i also goes to work”) attivano la classe **SERIOUS** (Figura 3.9), segnalando la necessità di attenzione prioritaria.

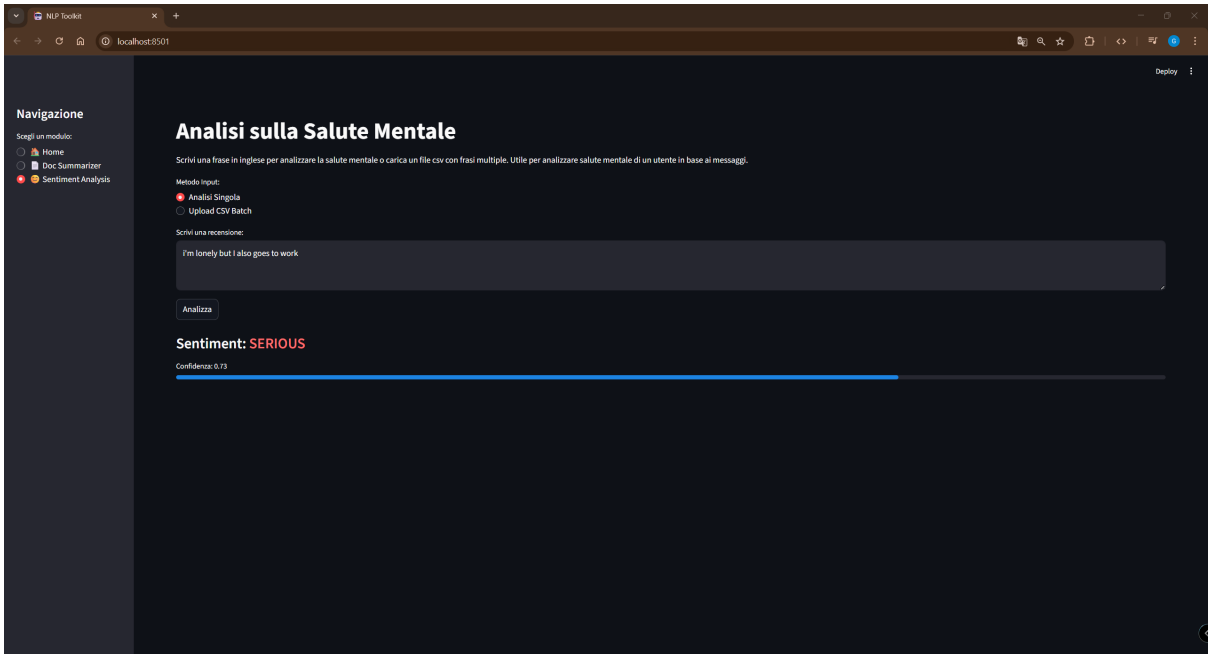

Figura 3.9: Identificazione di stati di disagio severo (Classe: SERIOUS).

Per analisi su larga scala, è disponibile la modalità ”Upload CSV Batch”. L’interfaccia (Figura 3.10) permette il caricamento di dataset preesistenti.

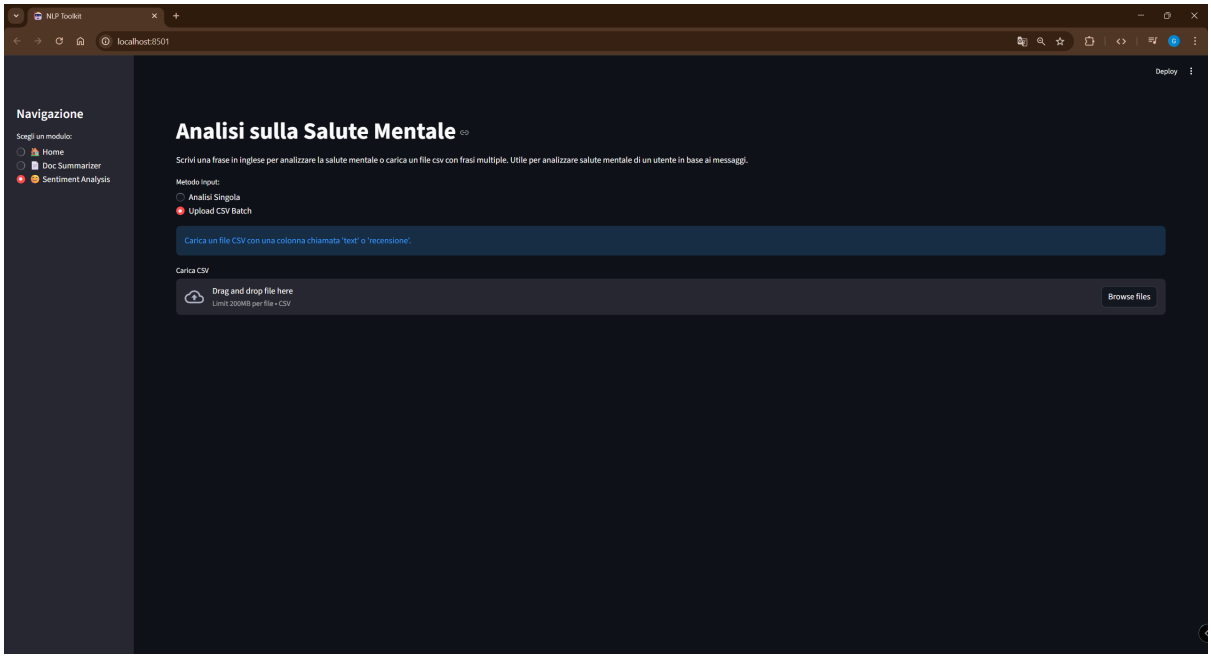

Figura 3.10: Interfaccia per il caricamento batch di file CSV.

Il sistema offre un’anteprima dei dati caricati per permettere all’utente di selezionare la colonna contenente il testo da analizzare (Figura 3.11).

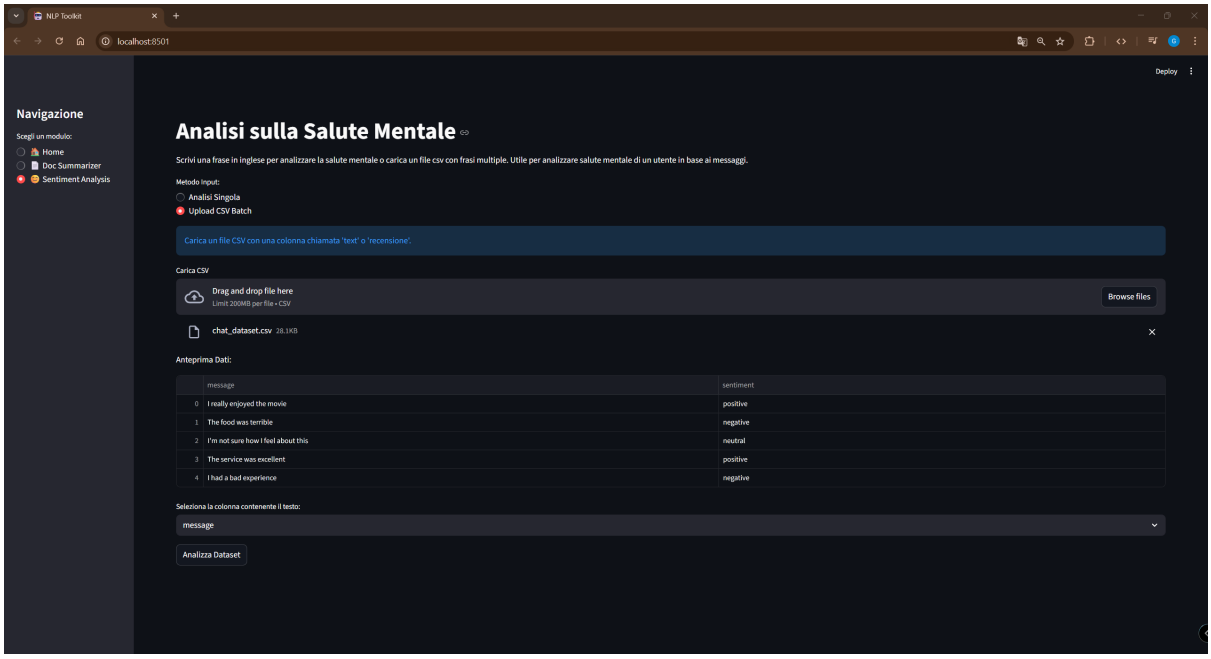

Figura 3.11: Anteprima dei dati CSV e selezione della colonna target.

Al termine dell'elaborazione, i risultati vengono aggregati e visualizzati tramite grafici interattivi. La Figura 3.12 mostra la distribuzione delle classi predette sull'intero file caricato, offrendo una panoramica immediata dello stato emotivo prevalente nel campione analizzato.

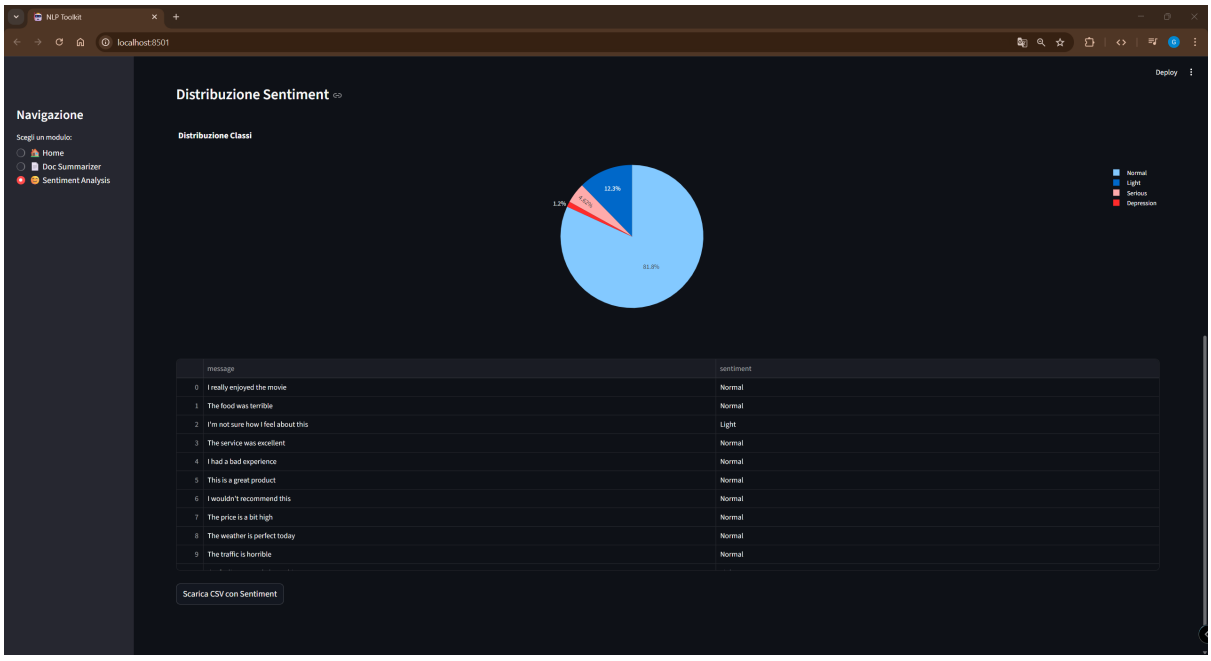

Figura 3.12: Visualizzazione aggregata dei risultati dell'analisi batch.

3.5 Metriche e Risultati Conclusivi

La valutazione quantitativa ha confermato la validità dell'architettura scelta, sia per quanto riguarda la capacità di sintesi che per la precisione nella classificazione.

3.5.1 Valutazione Summarization (ROUGE)

Per la sintesi, abbiamo utilizzato le metriche ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*), confrontando i riassunti generati con quelli di riferimento prodotti manualmente. La Figura 3.13 mostra i risultati ottenuti su campioni tematici specifici.

```
================================================
QUANTITATIVE EVALUATION (ROUGE SCORES)
================================================
Metric                                  Score
--------------------------------------  -------
ROUGE-1 (Unigram overlap)               31.25%
ROUGE-2 (Bigram overlap)                11.33%
ROUGE-L (Longest Common Subsequence)    25.82%
```

Figura 3.13: Metriche ROUGE per task di summarization

I valori ottenuti indicano una buona capacità del modello di catturare i contenuti informativi principali (ROUGE-1). Inoltre, il punteggio ROUGE-L suggerisce che il modello genera sequenze che rispettano in modo soddisfacente la struttura sintattica dei riassunti di riferimento, benché la natura astrattiva del task comporti inevitabilmente variazioni lessicali che penalizzano le metriche basate su n-grammi.

3.5.2 Valutazione Classificazione

Per la classificazione della salute mentale, il confronto tra le architetture testate ha evidenziato in maniera netta la superiorità dell'approccio Transformer rispetto alle tecniche precedenti.

Di seguito un confronto delle prestazioni dei due modelli su uno stesso set di test.

```
LSTM - Accuracy: 0.7260
              precision    recall  f1-score   support

  depression      0.662     0.544     0.597       259
       light      0.690     0.774     0.730       190
      normal      0.880     0.921     0.900       278
     serious      0.643     0.667     0.655       273

    accuracy                          0.726      1000
   macro avg      0.719     0.726     0.720      1000
weighted avg      0.723     0.726     0.722      1000


BERT (xlm-roberta + LoRA) - Accuracy: 0.7780
              precision    recall  f1-score   support

  depression      0.723     0.707     0.715       259
       light      0.767     0.884     0.822       190
      normal      0.945     0.863     0.902       278
     serious      0.682     0.685     0.684       273

    accuracy                          0.778      1000
   macro avg      0.779     0.785     0.781      1000
weighted avg      0.782     0.778     0.779      1000
```

Figura 3.14: Metriche dei modelli: il BERT è risultato superiore sotto ogni punto di vista rispetto l'LSTM.

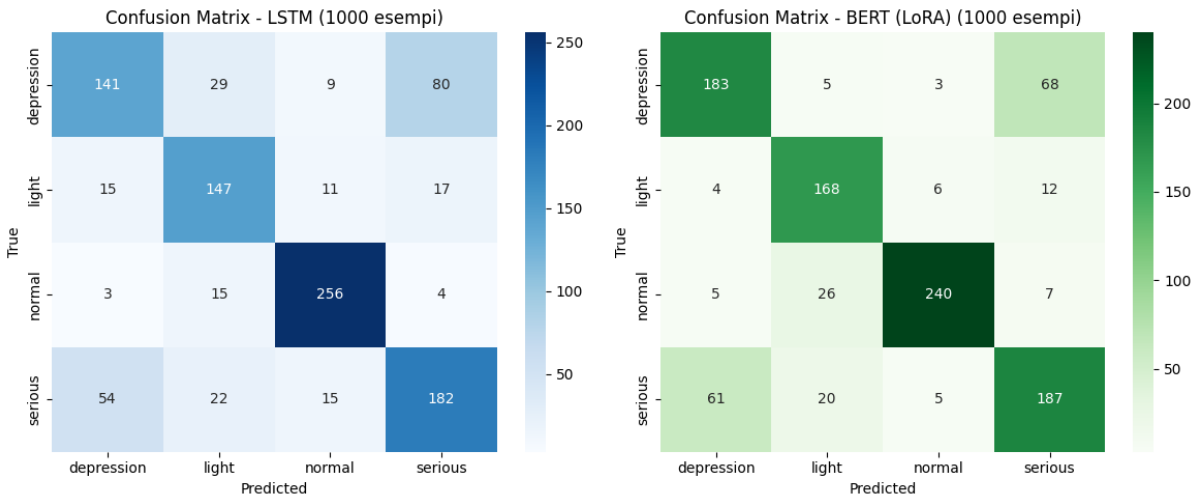

Figura 3.15: Metriche dei modelli: matrici di confusione dei due modelli.

L'utilizzo di XLM-RoBERTa con adapter LoRA ha portato a un miglioramento di circa 10 punti percentuali sull'accuracy rispetto alla rete LSTM e di 20 punti rispetto alla baseline.

3.5.3 Limitazioni e Sviluppi Futuri

Nonostante i risultati positivi, il sistema presenta alcune limitazioni attuali che verranno indirizzate nei prossimi sviluppi:

- **Gestione PDF**: l'attuale sistema di estrazione si basa su livelli testuali; documenti scansionati o immagini richiederebbero un'integrazione OCR (es. Tesseract).
- **Lingua del modello Mental Health**: il modello è attualmente addestrato su dati in inglese; per l'italiano sarebbe necessario un fine-tuning su un dataset specifico non ancora integrato.
- **Persistenza della sessione**: l'applicazione Streamlit non mantiene lo stato tra le sessioni (es. cronologia dei riassunti effettuati).

Gli sviluppi futuri prevedono l'integrazione di un sistema RAG (Retrieval-Augmented Generation) on-the-fly per permettere domande puntuali sui documenti (Q&A) e l'esportazione dei report finali in formato PDF scaricabile; attualmente è in sviluppo nella repository *ALLMond* su GitHub. In conclusione, il progetto Faboulous-Interpretr dimostra l'efficacia dell'uso di tecniche PEFT per portare capacità NLP avanzate su architetture hardware accessibili.