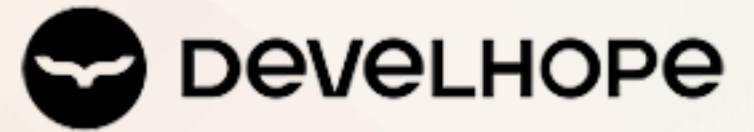


# Il valore delle opinioni

**Sentiment Analysis  
per e-commerce**

**Team:**

Francesca Amarena  
Matteo Gagliardi



# Progetto: Il Valore delle opinioni



**Scegliere un prodotto  
può essere  
complesso.**



**Interpretare le  
opinioni lo rende più  
semplice.**

**Team:**

Francesca Amarena  
Matteo Gagliardi

**Tutor:**

Pier Giorgio Poloni



# Obiettivi del progetto

## Obiettivo principale:

Costruire un modello di Sentiment Analysis in grado di analizzare una frase in Linguaggio Naturale e classificare automaticamente le recensioni in tre classi:

- Negativo
- Neutro
- Positivo

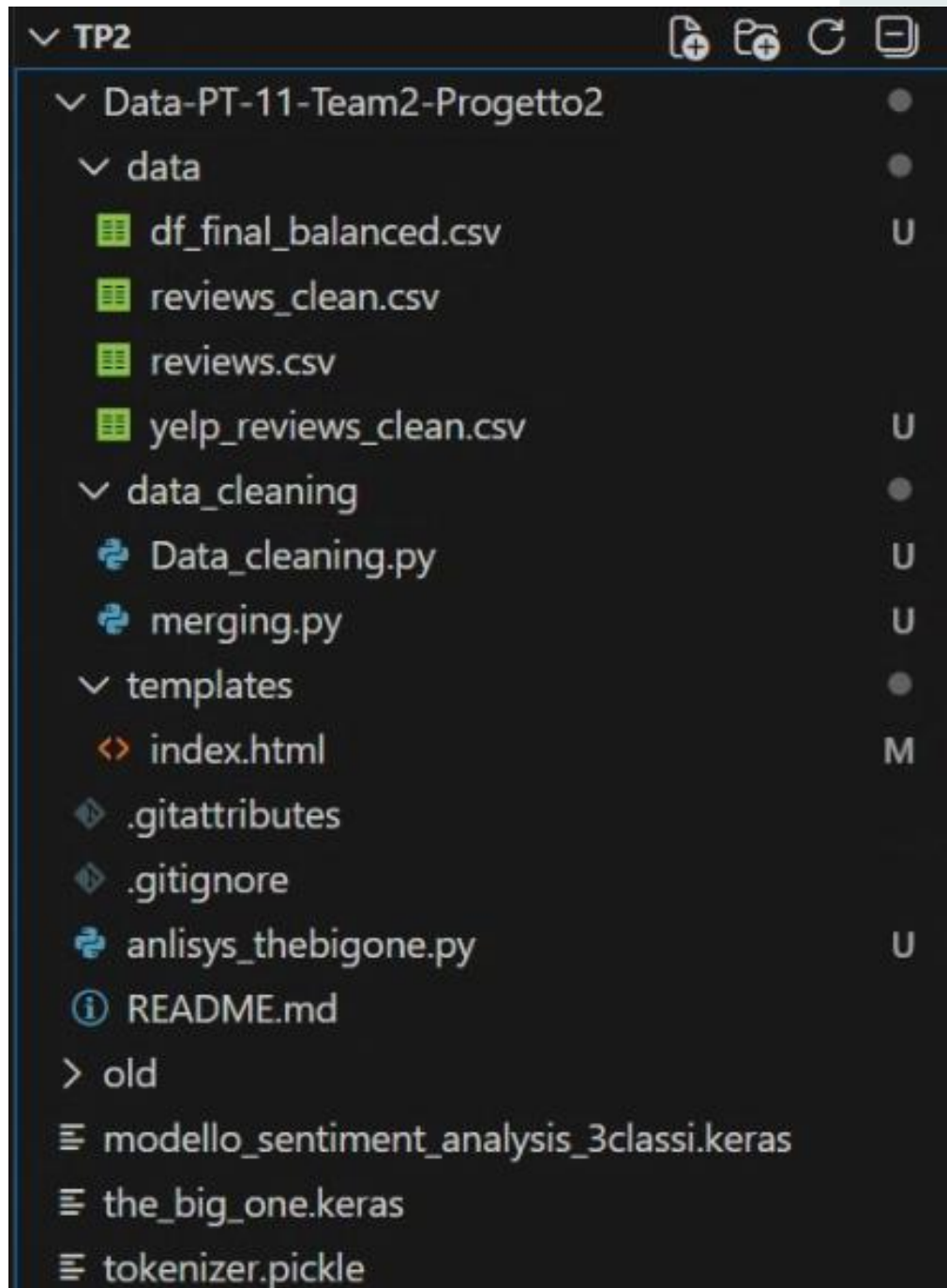
## Obiettivo secondario:

Testare il modello su dati mai visti

Rendere il modello utilizzabile tramite una semplice interfaccia web

# Dataset utilizzati

Dataset	Fonte	Ruolo	Motivazione della scelta
Amazon Reviews	Kaggle	Training	<ul style="list-style-type: none"><li>• Recensioni reali</li><li>• Rating 1-5</li><li>• Colonna testo</li></ul>
Yelp Reviews	Yelp Open Data	Generalizzazione	<ul style="list-style-type: none"><li>• Rating 1-5</li><li>• Colonna testo</li></ul>



# Struttura del Progetto

Abbiamo suddiviso il progetto come segue:

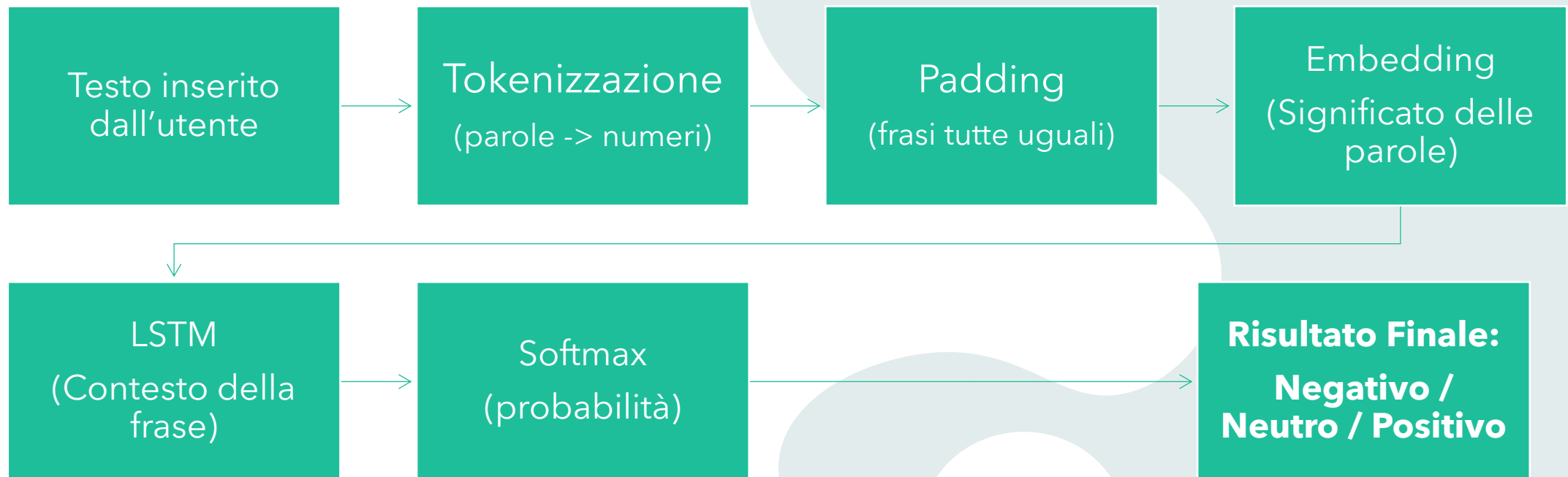
- La cartella **Data** contiene I file csv
- La cartella **Cleaning** contiene i file di pulizia dei dataset ed il file di merging
- La cartella **Templates** contiene l'HTML chiamato dall'app.py
- La macro-cartella **DATA-PT** contiene all'interno, oltre le suddette, anche i file di funzionamento del modello

# Data Cleaning & Preprocessing

Pulizia del testo	Sentiment Analysis	Unione dataset	Interfaccia web
Rimozione valori nulli e duplicati	<b>Tokenizzazione</b> -> spezza il testo e lo trasforma in numeri	Concatenazione tra i due dataset	HTML per il frontend
Normalizzazione del testo	<b>Embedding</b> -> da un valore numerico alle parole	Riduzione testo per addestrare modello	Flask per il Backend
Rimozione stopwords	<b>LSTM</b> -> capisce il senso della frase leggendo le parole in sequenza		Chiamata al modello

# Il modello

Il testo viene trasformato in numeri, analizzato nel suo contesto e classificato automaticamente in base al sentiment.



# MODELLO

---

## **Addestramento del modello:**

- Dataset di training pulito
- Classificazione multiclasse (negativo / neutro / positivo)

---

## **Testing**

- Valutazione delle performance sui dati di training
- Test del modello su una lista all'interno del codice
- Verificare generalizzazione e affidabilità



# Difficoltà durante l'addestramento

## 1. Addestramento con grandi quantità di dati

Necessità di :

- Pulizia dei dati
- Bilanciamento delle classi
- Gestione del rumore

(Non è sufficiente avere molti dati: devono essere **puliti, coerenti e rappresentativi.**)

## 2. Difficoltà nel riuscire a «scovare» il valore Neutrale, perché le parole usate nel neutrale sono facilmente confondibili

Esempio:

- The product is okay -> neutro
- The product is not bad -> positivo, ma linguisticamente ambiguo

## Sentiment Classification

● Online

Enter a sentence to analyze its sentiment.

Sentence

◆ Example: This course was very useful

**Analyze**

Tip: longer sentences usually give more accurate results.

# Interfaccia Web

# Interfaccia WEB

## Obiettivo:

Realizzare una **web application** che consenta all'utente di inserire una frase in linguaggio naturale e ottenere una **classificazione del sentiment** in tre classi

## Architettura dell'applicazione:

**Frontend:** pagina web con form di input testuale

**Backend:** Flask (Python)

**Modello AI:** TensorFlow / Keras

## Tecniche NLP:

- Tokenizzazione del testo
- Padding delle sequenze
- Classificazione multi-classe

```
{% if classificazione %}
<div class="result">
  <p class="k">Input</p>
  <p class="v">“{{ frase }}”</p>

  <p class="k" style="margin-top:14px;">Result</p>
  <div class="badge"><i></i> {{ classificazione }}</div>
</div>
{% endif %}
```

```
if request.method == 'POST':
    frase = request.form['frase']

    sequenze = tokenizer.texts_to_sequences([frase])          # Trasformiamo la frase in sequenze
    sequenze_padded = pad_sequences(
        sequenze,
        maxlen=lungh_sequenze,
        padding='post',
        truncating='post'
    )

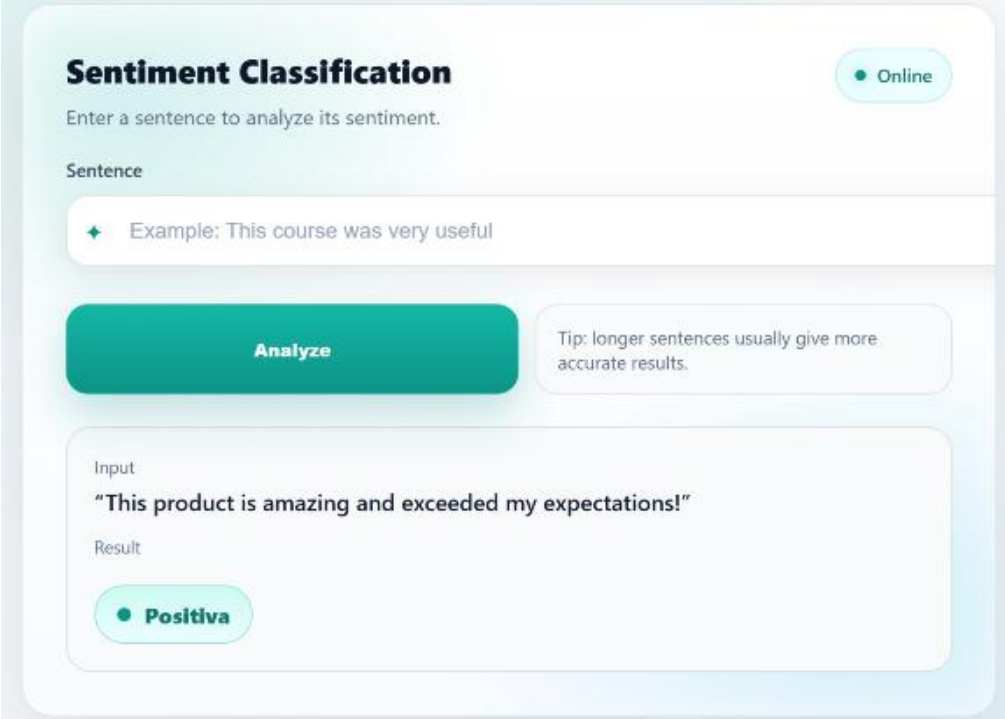
    pred = model.predict(sequenze_padded)                    # shape (1,3)
    proba = pred[0]                                          # es: [0.2, 0.3, 0.5]
    classe = int(np.argmax(proba))                          # 0/1/2 (nello specifico prende l'indice della p
    # Mappa numero → etichetta testuale
    mappa = {0: "Negativa", 1: "Neutrale", 2: "Positiva"}
    # Recuperiamo la classificazione finale
    classificazione = mappa[classe]

    # Questo rimanda i dati alla pagina html
    return render_template('index.html', frase=frase, classificazione=classificazione)
```

# Dimostrazione Recensione Positiva

Frase analizzata: "This product is amazing and exceded my expectations!"

Risultato: "Positiva"



The screenshot shows a web application titled "Sentiment Classification" with a toggle switch set to "Online". Below the title, it says "Enter a sentence to analyze its sentiment." There is a "Sentence" input field containing an example: "Example: This course was very useful". A green "Analyze" button is positioned below the input field. To the right of the button, a tip box states: "Tip: longer sentences usually give more accurate results." Below the "Analyze" button, the "Input" section displays the sentence: "This product is amazing and exceded my expectations!". The "Result" section shows the output: "Positiva", which is highlighted in a green pill-shaped button.

**Sentiment Classification** Online

Enter a sentence to analyze its sentiment.

Sentence

Example: This course was very useful

**Analyze**

Tip: longer sentences usually give more accurate results.

Input

"This product is amazing and exceded my expectations!"

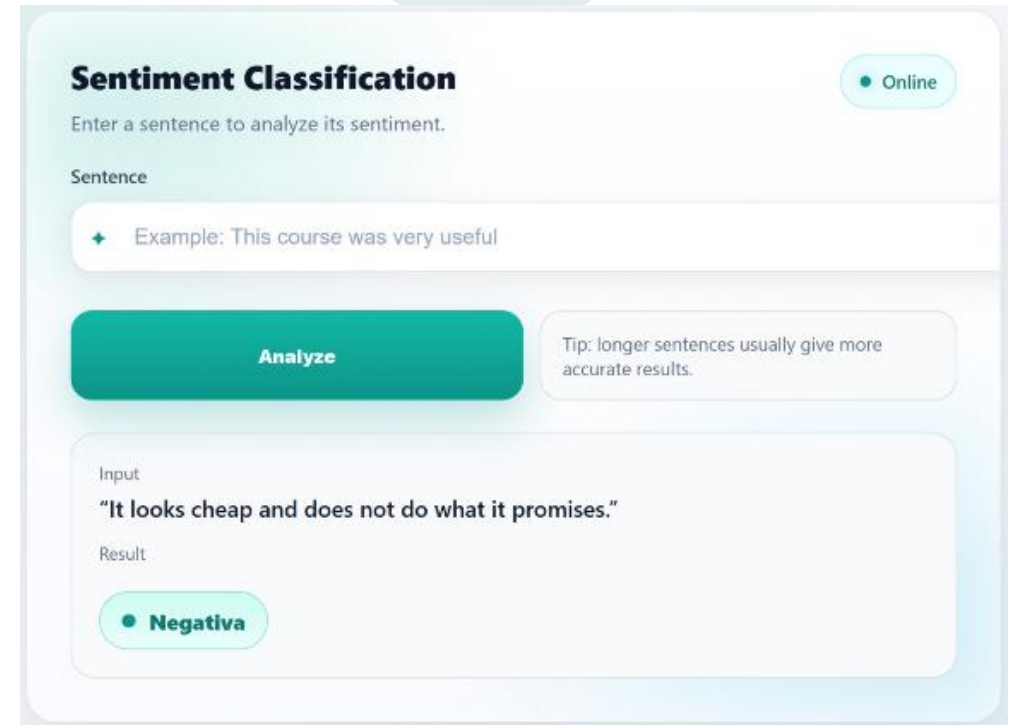
Result

**Positiva**

# Dimostrazione Recensione Negativa

Frase analizzata: "It looks cheap and does not do what it promise"

Risultato: "Negativa"



The image shows a web application for sentiment classification. At the top, it says "Sentiment Classification" with an "Online" status indicator. Below this, it prompts the user to "Enter a sentence to analyze its sentiment." There is a text input field with an example sentence: "Example: This course was very useful". A large green "Analyze" button is positioned below the input field. To the right of the button, a tip states: "Tip: longer sentences usually give more accurate results." Below the input field, the "Input" section displays the sentence: "It looks cheap and does not do what it promises." The "Result" section shows the classification: "Negativa".

**Sentiment Classification** Online

Enter a sentence to analyze its sentiment.

Sentence

+ Example: This course was very useful

**Analyze**

Tip: longer sentences usually give more accurate results.

Input

"It looks cheap and does not do what it promises."

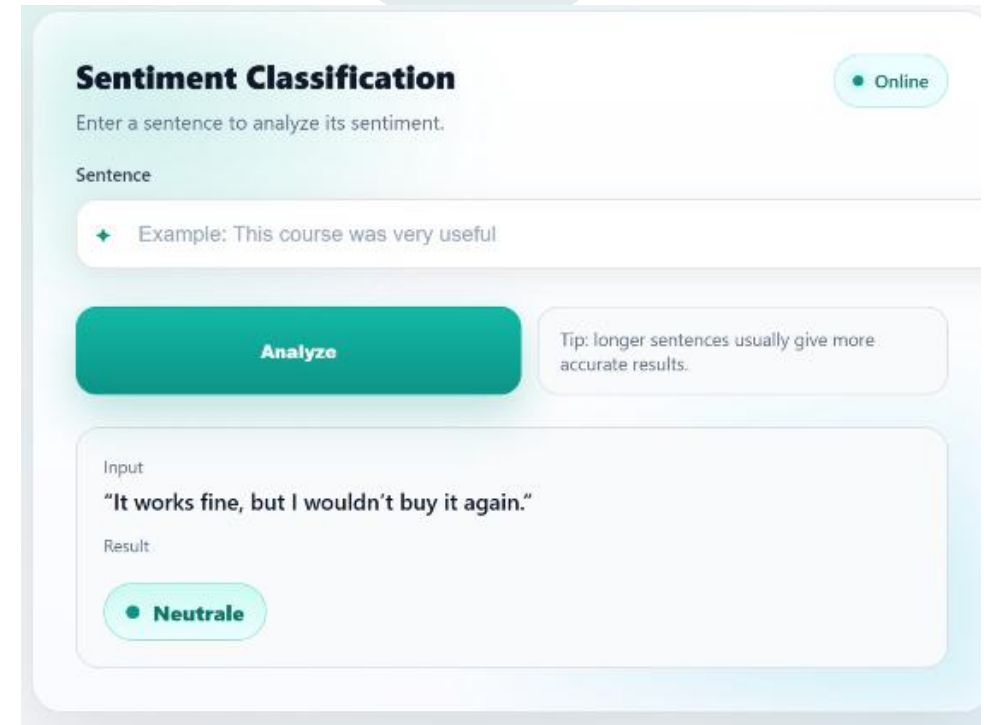
Result

**Negativa**

# Dimostrazione Recensione Neutrale

Frase analizzata: "It works fine, but I wouldn't buy it again"

Risultato: Neutrale



The screenshot shows a web application titled "Sentiment Classification" with a toggle switch set to "Online". Below the title, it says "Enter a sentence to analyze its sentiment." There is a text input field labeled "Sentence" containing the example text "Example: This course was very useful". A green "Analyze" button is positioned below the input field. To the right of the button, a tip box states: "Tip: longer sentences usually give more accurate results." Below the input field, the "Input" section displays the sentence "It works fine, but I wouldn't buy it again." The "Result" section shows a green button with the word "Neutrale" and a small green dot to its left, indicating the sentiment classification.

# Conclusioni

Il progetto dimostra come la Sentiment Analysis possa trasformare feedback testuale non strutturato in **insight utili e azionabili**.

Il modello sviluppato è in grado di analizzare nuove recensioni in modo affidabile e rappresenta una soluzione valida e veloce per interpretare opinioni e/o recensioni nel campo dell' e-commerce.

# SVILUPPI FUTURI

Miglioramento grafico dell'interfaccia web con inserimento dell'accuratezza del modello (matplotlib)

Riconoscimento di più lingue

Aumentare il riconoscimento delle sfumature delle recensioni



# **GRAZIE PER L'ATTENZIONE**

**Corso: Data & AI Analyst**

