Chatbot Morano Calabro Travel

# Descrizione

Questo progetto implementa un chatbot turistico per il comune di Morano Calabro, in Italia. Il chatbot è progettato per rispondere a domande comuni su argomenti come alloggi, ristoranti, attrazioni, eventi, attività all'aperto e informazioni pratiche.

Il chatbot utilizza reti neurali ricorrenti (LSTM) per comprendere l'intento dell'utente e fornire risposte adeguate. Il modello è addestrato su dati di testo che rappresentano vari intenti e risposte tipiche, permettendo al chatbot di interpretare e rispondere a domande in linguaggio naturale.

# Struttura del Codice

## Caricamento e Configurazione dei Pacchetti

Importa librerie di machine learning (tensorflow, keras), manipolazione dati (numpy), visualizzazione (matplotlib, seaborn), database (sqlite3), e NLP (nltk).

## Preparazione dei Dati

Definisce un dizionario di **intenti** e **risposte** che comprende vari scenari di interazione (es. saluto, prenotazione, consigli turistici).

Crea una tabella interactions in un database **SQLite3** per salvare le interazioni con gli utenti.

## Tokenizzazione e Pre-elaborazione del Testo

Utilizza il **Tokenizer** di tensorflow.keras per convertire i dati di testo in sequenze di token, gestendo le parole sconosciute con un token OOV (<OOV>).

Applica il padding alle sequenze per garantire che abbiano tutte la stessa lunghezza.

## Creazione e Addestramento del Modello

Definisce un **modello sequenziale** con livelli come:

Embedding: per rappresentare le parole come vettori numerici.

LSTM bidirezionale: per catturare le dipendenze nel testo in entrambe le direzioni.

Dense con relu e softmax per la classificazione degli intenti.

Compila e addestra il modello per **300 epoche**, usando sparse\_categorical\_crossentropy come funzione di perdita e Adam come ottimizzatore.

## Funzione di Visualizzazione

Implementa plot\_metrics() per visualizzare l'**andamento della loss e accuracy** durante l'addestramento, salvando il grafico come immagine model\_metrics.png.

## Funzioni di Preprocessing e Predizione

**Preprocesso del Testo**: pulisce l'input dell'utente con espressioni regolari.

**Correzione Ortografica** e **Estrazione di Parole Chiave**: funzioni placeholder attualmente non operative.

**Predizione Intento**: utilizza il modello addestrato per predire l'intento dell'utente basato sull'input preprocessato.

## Ciclo Principale del Chatbot

Il chatbot accetta input testuali dall'utente, utilizza la funzione di predizione dell'intento per identificare l'argomento, e risponde con una frase adeguata scelta casualmente dal dizionario delle risposte.

Registra ogni interazione nel database SQLite.

## Richiesta di Feedback

Alla fine dell’interazione, il chatbot chiede un feedback all'utente sulla soddisfazione delle risposte, salvandolo nel database.

## Esplorazione Dati delle Interazioni

La funzione show\_interaction\_data() stampa i dati salvati delle interazioni, utili per analizzare le prestazioni del chatbot.

## Chiusura della Connessione

La connessione al database SQLite viene chiusa per assicurare l’integrità dei dati.