



Università degli studi di Bari Aldo Moro

Dipartimento di Informatica
Corso di Laurea in Informatica

Tesi di Laurea in Informatica

Guida utente dell'applicativo realizzato

Professore

Prof. Stefano Ferilli

Relatrice

Prof. Berardina Nadja De Carolis

Laureando

Alessandro Carella

Anno Accademico 2022 - 2023

Indice

1	Guida all'installazione e all'utilizzo	1
1.1	Introduzione	1
1.2	Installazione	1
1.2.1	Programmi necessari	1
1.2.2	Installazione pacchetti necessari	2
1.3	Avvio del programma	3
1.4	Descrizione dell'interfaccia	6

Capitolo 1

Guida all'installazione e all'utilizzo

1.1 Introduzione

In concomitanza alla realizzazione del testo di laurea è stato realizzato un applicativo che permette di identificare, in tempo reale, il mood della persona che lo sta utilizzando.

Ho ritenuto opportuno realizzare questo applicativo in modo da poter avere un riscontro diretto del lavoro prodotto.

È difatti possibile selezionare fra tutti i modelli predittivi realizzati ed utilizzare quello scelto per effettuare le predizioni in tempo reale sulla persona che sta utilizzando il programma.

1.2 Installazione

1.2.1 Programmi necessari

È possibile scaricare l'applicativo direttamente dalla repository github dedicata alla tesi, presente a questo [link](#).

Una volta scaricato il repository sarà necessaria l'installazione del compilatore python:

- Questo è il [link](#) per scaricare il compilatore dal microsoft store, preferibile se si sta utilizzando un computer con sistema operativo windows.
- Questo invece è il [link alternativo](#) alla pagina ufficiale di download di python, si consiglia di scaricare la versione 3.9 in quanto è la versione utilizzata durante lo sviluppo.

Una volta scaricato il compilatore si consiglia di scaricare un IDE (Integrated development environment) come [pycharm](#) o un text editor come [Visual Studio Code](#).

Per lo sviluppo è stato utilizzato Visual Studio Code con la relativa [estensione](#) che permette di eseguire i file all'interno del programma.

1.2.2 Installazione pacchetti necessari

Per funzionare il programma ha bisogno dell'installazione di diversi pacchetti o packages installabili direttamente attraverso pip (installato insieme a python).

Per installare questi è possibile aprire un terminale (prompt dei comandi o windows powershell per esempio) all'interno della cartella scaricata da github ed eseguire il comando:

```
pip install -r requirements.txt
```

È necessario un ulteriore pacchetto non compreso all'interno del file requirements.txt in quanto dipende dalla macchina sulla quale viene eseguita l'installazione del programma:

Questo pacchetto è py-torch e rimando a [questa](#) guida per l'installazione di questo in quanto sono presenti diversi fattori nella scelta del comando di installazione giusto e trattarli nella guida risulterebbe eccessivamente complicato, specialmente per la scelta della versione CUDA da installare; faccio notare però che, per quanto la guida al link precedente descriva il processo di installazione tramite conda, non è possibile seguire questo processo da questa guida in quanto non è stato installato questo package manager, mentre è stato installato pip; basterà quindi ignorare i passaggi strettamente relativi a conda e selezionare “pip” come, ad esempio, nell'immagine 1.1.

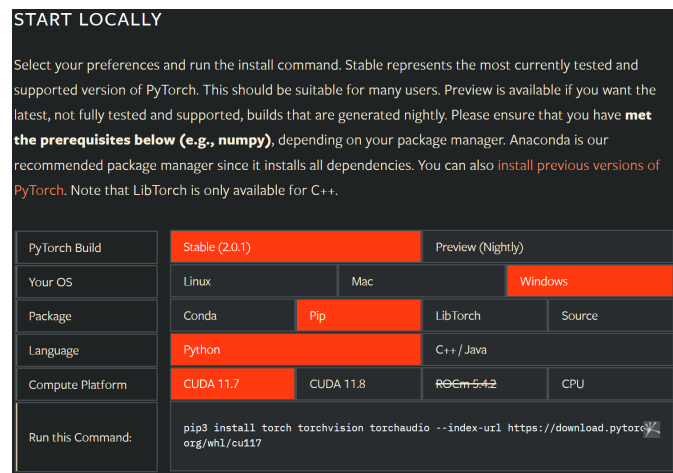


Figura 1.1: Pagina per l'installazione di py-torch

È importante notare che, per quanto non sia impossibile eseguire il programma sprovvisto di questa tecnologia, è preferibile installare il programma utilizzando la tecnologia CUDA, in quanto permette di ottenere delle performance migliori.

1.3 Avvio del programma

Una volta installati tutti i programmi e i pacchetti necessari per l'esecuzione del programma basterà eseguire il file `interfaccia.py` presente nella cartella `.../bachelor-s-thesis/src/sviluppo` con l'IDE o il text editor scelto.

Di seguito una procedura più dettagliata per l'esecuzione attraverso Visual Studio Code:

- Aprire il programma Visual studio code, come visibile in 1.2

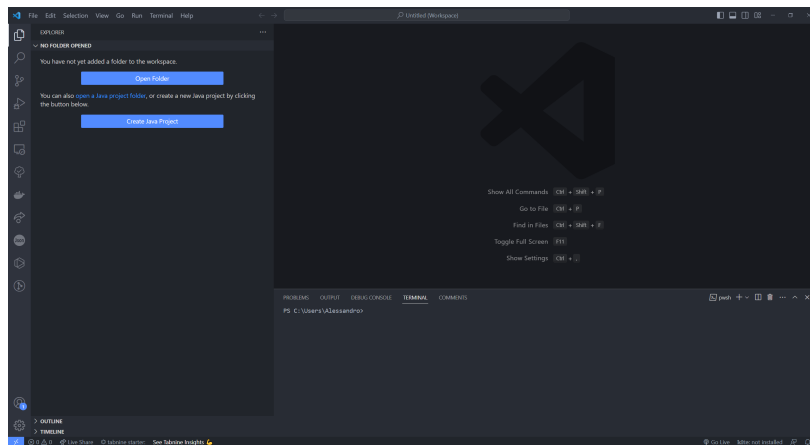


Figura 1.2: Apertura di Visual Studio Code

- Premere il bottone “Open Folder” e selezionare la cartella scaricata da github, come visibile in 1.3

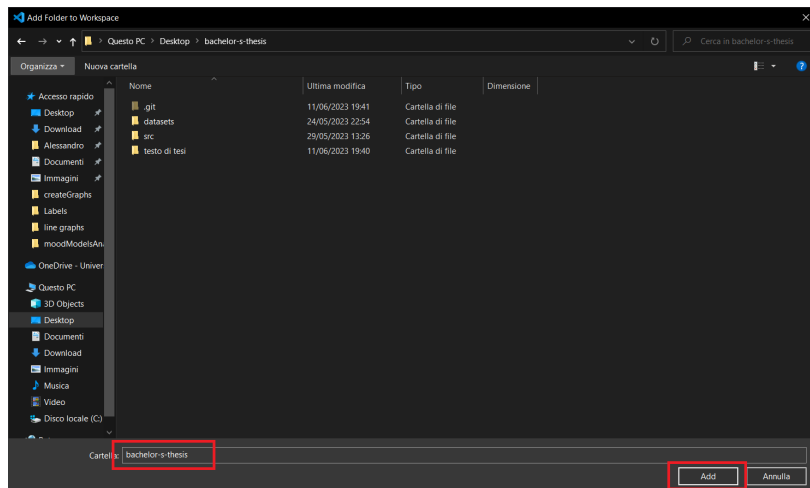


Figura 1.3: Apertura cartella della repository

- Ora premere sull'icona di "Run and Debug" presente sulla sinistra e premere sul bottone "Run and Debug", come visibile in 1.6

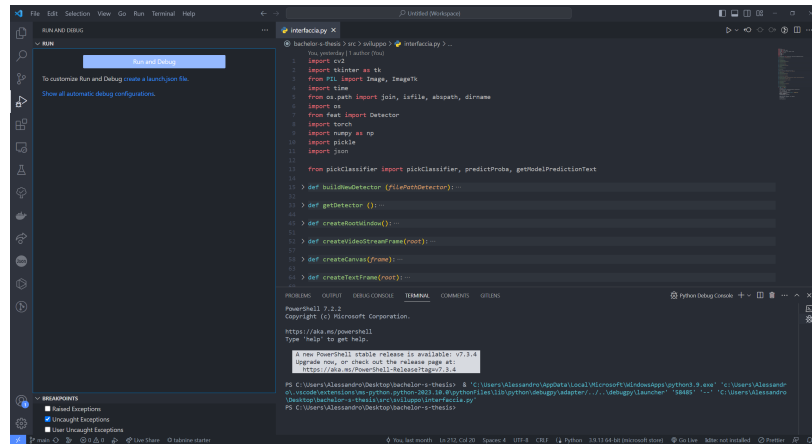


Figura 1.6: Run and Debug

- Una volta premuto il bottone si presenterà questo menu dal quale si consiglia di scegliere la prima opzione, "Python File", come visibile in 1.7

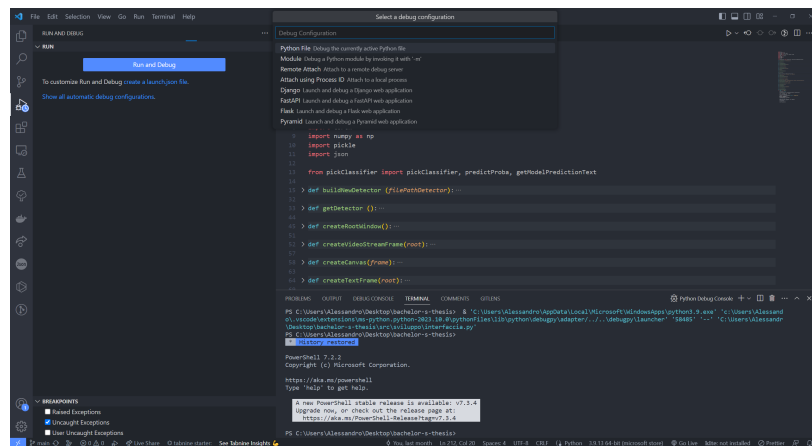


Figura 1.7: Selezione "Python File"

- Ora il programma si avvierà e la prima schermata presente sarà quella visibile in 1.8

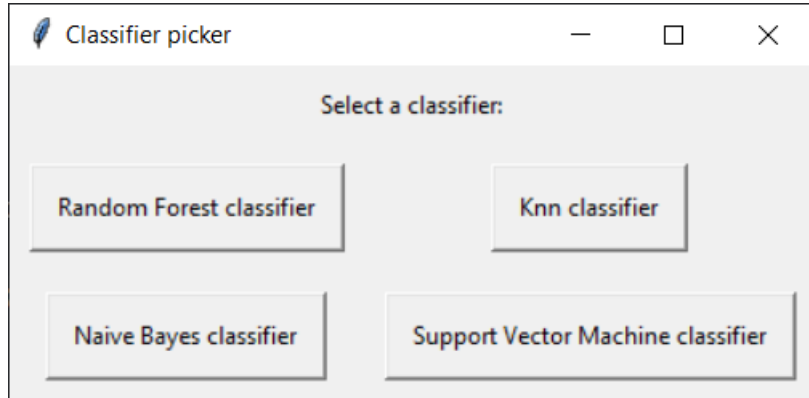


Figura 1.8: Prima schermata

da qui è possibile scegliere fra i 4 classificatori creati per il progetto, si consiglia di scegliere, premendo il relativo bottone, il “Random Forest classifier” in quanto risulta essere il più performante fra i quattro.

- Una volta selezionato il classificatore si presenterà l'interfaccia finale dell'applicativo.

1.4 Descrizione dell'interfaccia

L'interfaccia presente è quella visibile in 1.9

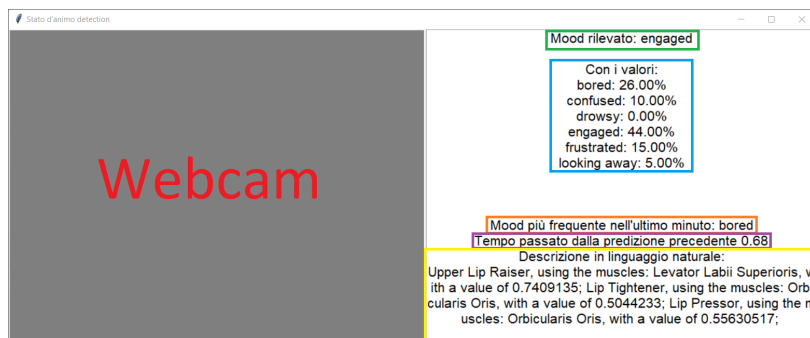


Figura 1.9: Schermata principale dell'interfaccia

- Nella metà a sinistra è presente lo stream della videocamera dalla quale vengono estratte le informazioni

- nella sezione in verde è presente il mood rilevato nel frame letto nel momento della visualizzazione
- nella sezione in blu è presente l'intera predizione effettuata dal modello predittivo (il mood mostrato nella sezione in verde è ovviamente il mood con il valore maggiore fra questi)
- nella sezione in arancione è presente il mood che è stato più frequente nell'ultimo minuto in modo che si possa avere un'idea più precisa del mood della persona che sta utilizzando il sistema
- nella sezione in viola è presente, in secondi, il tempo passato dalla predizione precedentemente effettuata, questo campo serve per effettuare una veloce valutazione delle performance del sistema sulla macchina
- nella sezione in giallo è presente la descrizione in linguaggio naturale delle Action Units rilevate, questa si compone di:
 - il nome associato all'action units che risulta attiva in quel momento.
 - il muscolo relativo contratto
 - il valore rilevato per l'Action Unit

Tutte le descrizioni delle action units sono presenti in sequenza separate da un punto e virgola.