**UrbanSecurity**

Progetto di Ingegneria della Conoscenza

A.A. 2021/2022

**Componenti del gruppo:**

* *Salvatore Napoli*

Matricola: 706483 Mail: s.napoli3@studenti.uniba.it

* *Pietro Terrone*

Matricola: 726944 Mail: p.terrone2@studenti.uniba.it

**Link del repository**:

**Documentazione**

**Introduzione**

Il progetto consiste in un’applicazione chiamata *UrabnSecurity*, una tecnologia di rilevamento della violenza in tempo reale dallo stream di una telecamera ma che può essere facilmente integrata con qualsiasi sistema di sicurezza. La sua funzione principale è garantire la sicurezza pubblica attraverso la sorveglianza visiva della folla, quindi qualsiasi attività violenta genera un allarme che automaticamente allerta le autorità.

Attraverso l'applicazione di tecniche di elaborazione delle immagini e apprendimento automatico, i sistemi di sorveglianza intelligenti sono in grado di estrarre e interpretare le informazioni dai filmati CCTV più velocemente e in modo molto più efficiente di qualsiasi osservatore umano. Inoltre, tutti i componenti lavorano insieme in un ambiente flessibile in modo da poter essere personalizzati per servire uno scopo specifico che può anche cambiare nel tempo.

Le feature che sono state inserite sono:

* *Rilevamento di persone e riconoscimento scene di violenza:* il sistema prende in input lo stream di una telecamera di sorveglianza, lo elabora e restituisce se vi sono presenti scene di violenza.
* *Predizione di violenza*: il sistema cerca di predire, attraverso vari fattori, i posti in cui si possono verificare scene di violenza.
* *Ricerca in un grafo:* il sistema calcola il percorso migliore, dato un punto di partenza (una stazione di polizia o la posizione esatta di una volante), per raggiungere la posizione della telecamera che ha lanciato l'allarme.
* *Interrogare una base di conoscenza:* il sistema permette di interrogare la KB per ottenere informazioni

-->(ad esempio: elenco lezioni del lunedì, elenco lezioni di un determinato professore, edificio in cui si trova un’aula…).

* **Rilevamento di persone e riconoscimento scene di violenza**

Una rete neurale convoluzionale è stata addestrata per il rilevamento della violenza in tempo reale.

Per il riconoscimento di persone nel video è stato usato Inception v3 è un modello di riconoscimento delle immagini ampiamente utilizzato che ha dimostrato di raggiungere una precisione superiore al 78,1% sul set di dati.

Dopo aver creato e addestrato il modello utilizzando l'apprendimento supervisionato, usando un ampio set di video etichettati, il sistema è in grado di elaborare in tempo reale il filmato e di restituire in output se vi è presente violenza nel video e con quale probabilità.

In caso di violenza il sistema dovrà lanciare un allarme, dopo un'attenta e approfondita analisi, è stato scelto, che l'allarme sarà lanciato solo le probabilità supera il 70%.



* **Ricerca in un grafo**

Si è scelto di implementare questa funzionalità affinché un’ipotetica volante possa orientarsi, attraverso l’uso di una mappa, al raggiungimento della posizione esatta della telecamera che ha lanciato l'allarme.

Per la creazione della mappa, è stata selezionata la città di Bari.

Le telecamere sono state posizionate in diversi punti della città sia nelle zone del centro ma anche in zone periferiche e sono locate in piazze strade ma anche in vicoli.

Di seguito la posizione esatta di ciascuna telecamera:

* Via Sparano da Bari, 97, 70121 Bari BA
* Via Abate Giacinto Gimma, 70122 Bari BA
* Via Alessandro Manzoni, 70122 Bari BA
* Via Benedetto Cairoli, 2, 70122 Bari BA
* Via Nicolò Putignani, 70121 Bari BA
* Via Dante Alighieri, 24, 70121 Bari BA
* Via Andrea da Bari, 59, 70122 Bari BA
* Corso Benedetto Croce, 70125 Bari BA
* Corso Sidney Sonnino, 70121 Bari BA
* Corso A. de Gasperi, 429, 70125 Bari BA
* Via Bruno Buozzi, 3-13, 70132 Bari BA
* Viale Biagio Accolti Gil, 22, 70132 Zona Industriale BA
* V.le Francesco de Blasio, 70132 Zona Industriale BA
* Via Giambattista Bonazzi, 70122 Bari BA
* Molo S. Nicola, 1, 70121 Bari BA
* Piazza Aldo Moro, 13, 70123 Bari BA
* Lungomare Armando Perotti, 70126 Bari BA
* Piazza del Redentore, 230c, 70123 Bari BA
* Piazza Giulio Cesare, 70122 Bari BA
* Via della Resistenza, 154, 70125 Bari BA
* Piazza Umberto I, 1, 70121 Bari BA
* Piazza Mercantile, 70122 Bari BA
* Strada Casamassimi, 11, 70122 Bari BA
* Piazza del Ferrarese, 10, 70122 Bari BA
* Piazza S. Nicola, 13, 70122 Bari BA
* Piazza Armando Diaz, 70121 Bari BA

Per la struttura del grafo, è stato creato appositamente un file *csv* contenente le informazioni riguardanti i luoghi e le vie di Bari.

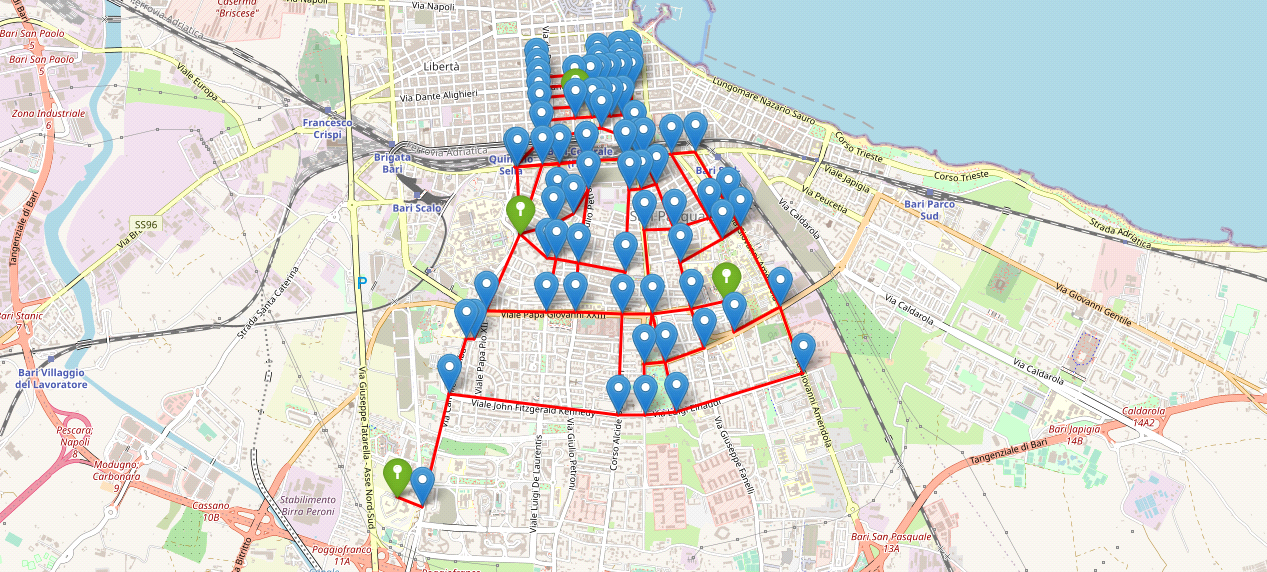


È stato utilizzato [*Google Maps*](https://www.google.it/maps) per prendere i riferimenti di inizio e fine di ognuna delle strade (talvolta suddivise in segmenti ridotti) che sono state mappate.   
Per ogni riga del dataset, sono state salvate:

* le coordinate (latitudine Y e longitudine X) del punto di partenza (X1, Y1), e del punto di arrivo (X2, Y2)
* nome della strada (Name)
* lunghezza della strada in metri (Length).

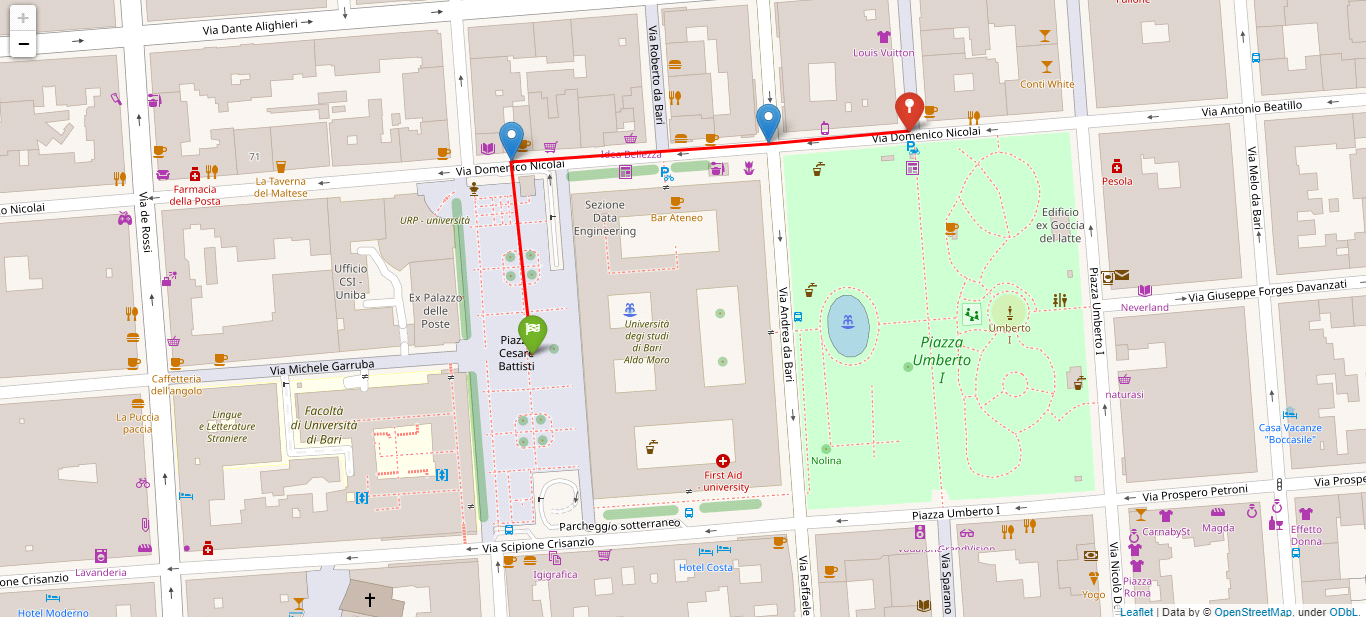
Dal [dataset](https://github.com/mmc185/YOUniversity/blob/main/resources/locations/Bari.csv) si è ricavato poi il grafo in cui ogni nodo rappresenta una coppia di coordinate (che identificano un unico luogo) e ogni arco una strada, con la sua lunghezza come peso.

Attraverso l’utilizzo della libreria folium, è stata applicata la parte di visualizzazione grafica sul servizio di [*OpenStreetMap*](http://www.openstreetmap.org/).



Per la ricerca su grafo, è stato implementato l’algoritmo **A\*** con funzione euristica calcolata mediante la *distanza euclidea* (in linea d’aria tra due punti).

Il risultato dell’algoritmo viene poi visualizzato sulla mappa tracciando in rosso il percorso e indicando con un’icona rossa il punto di partenza, verde l’obiettivo e i nodi intermedi in blu.



Contenuto del package *Pathfinding*:

* *GeoLocationsUtilities*: file contenente utilities per la gestione dei luoghi
* *GraphUtilies*: file contenente classi di supporto al grafo
* *PathFinding*: file di connessione tra la struttura del grafo e la mappa
* *SearchProblemUtilities*: file contenente utilities per effettuare la ricerca

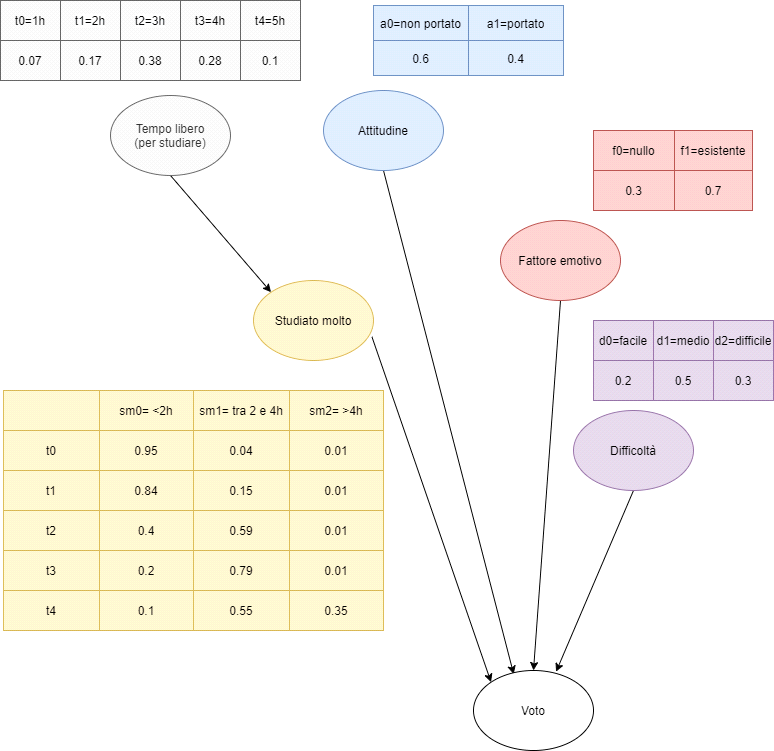
Librerie utilizzate:

* [Folium](https://python-visualization.github.io/folium/)

* **Predizione scene di violenza**

Per questa funzionalità si è creata una **Belief Network** (**Rete Bayesiana**) per predire, in base a dei fattori, dove c'è più probabilità che si verifichino scene di violenza.

La rete bayesiana costruita è la seguente:



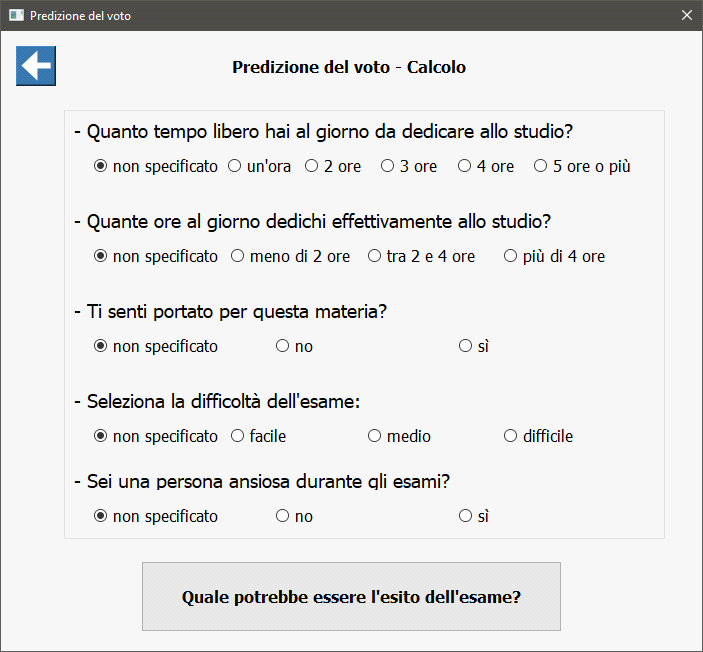
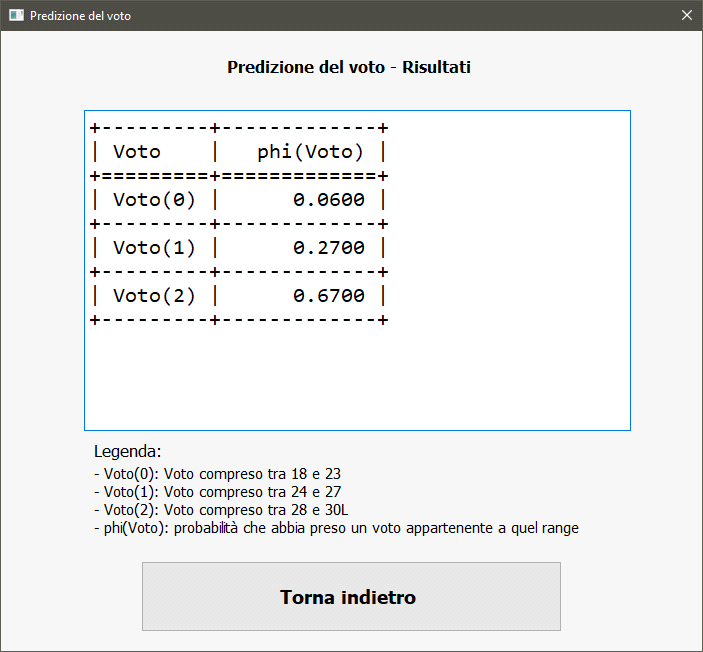
La visualizzazione testuale completa delle tabelle è presente nel file [*reteBayesiana.txt*](https://github.com/mmc185/YOUniversity/blob/main/doc/reteBayesiana.txt)

Come si evince dall’immagine, la probabilità che si verifichi violenza può essere( [Bassa], [Media], [Alta] ) ed è influenzata dalle seguenti features:

* *mole di persone presenti in quell'istante* [x < 150 persone,  
  150 persone ≤ x ≤ 600 persone, x > 600 persone]
* *orario* [notturno, diurno]
* *giorno*[festivo, feriale]
* luogo [vicolo, starda, piazza]
* influenzate, a loro volta, anche dalla *posizione*  [centro, periferia]

A chi utilizzerà il sistema è richiesto di indicare i valori di uno o più fattori e il sistema calcolerà la predizione di violenza in base alle probabilità di essi.

La libreria che implementa la rete bayesiana è *pgmpy*, che permette di creare le tabelle di distribuzione di probabilità e fare inferenza sulla predizione del voto, basandosi sulla struttura della Belief Network.

Contenuto del package *bayesianNetwork*:

* *bayesianNetwork*: file che contiene la creazione della Belief Network e la funzione per fare inferenza.

Librerie utilizzate:

* [pgmpy](https://pgmpy.org/)

* **Interrogare la base di conoscenza**

Un’altra funzionalità del sistema permette allo studente di interrogare la base di conoscenza al fine di ottenere varie informazioni, tra cui: orari delle lezioni, quali professori insegnano una certa materia o tutte le lezioni di un determinato giorno della settimana.

Per far ciò è stata usata la libreria *pytholog* che permetteva di popolare una base di conoscenza ed effettuare query su di essa.

Le query:

* Permettono di verificare se una clausola è conseguenza logica della base di conoscenza (in questo caso, il sistema risponderà “*Yes*”, altrimenti “*No*”).
* Possono comprendere variabili in modo che la risposta riporti tutti i valori che possono essere assunti dalla/e variabile/i.

*Esempio*: la query “*insegna(P,M)*” ridarà tutte le coppie ove *P* è un professore e *M* è la materia insegnata.

Gli assiomi utilizzati nella KB sono i seguenti (per la maggior parte, è riportato solo un esempio con relativa spiegazione) :

* insegna(fanizzi,icon) insegna(Professore, Materia)
* in(aula4,palazzo\_aule) in(Aula, Edificio)
* corso(di\_terlizzi,a1) corso(Professore, Anno\_corso)
* orario(di\_terlizzi,h9,lunedi) orario(Professore, Ora\_inizio, Giorno)
* lezione(M,H,G,P):-insegna(P,M),orario(P,H,G)   
  lezione(Materia, Ora\_inizio, Giorno, Professore):-insegna(Professore, Materia),orario(Professore, Ora\_inizio, Giorno)
* classe(a1,aula4) classe(Anno\_corso, Aula)
* luogo(C,X):-classe(C,A),in(A,X)   
  luogo(Anno\_corso, Palazzo):-classe(Anno\_corso, Aula),in(Aula, Palazzo)

Le informazioni salvate nella KB riguardano il primo semestre dei tre anni del corso di Informatica.

Contenuto del package *Prolog*:

* *prolog*: file che contiene la creazione della KB e la funzione per interrogarla.

Librerie utilizzate:

* [pytholog](https://mnoorfawi.github.io/pytholog/)

**Interfaccia grafica**

Per collegare le varie funzionalità, è stata creata un’interfaccia grafica molto semplice attraverso l’utilizzo del tool [*Qt HYPERLINK "https://github.com/altendky/pyqt-tools" De HYPERLINK "https://github.com/altendky/pyqt-tools"signer*](https://github.com/altendky/pyqt-tools).



Librerie:

* [PyQt5](https://www.riverbankcomputing.com/software/pyqt/)

**Sviluppi futuri**

Questo sistema può essere esteso in più modi, ad esempio: ampliando la KB, inserendo altri luoghi e/o strade sulla mappa o aggiungendo altre funzionalità.