UNIVERSITÀ
DEGLI STUDI
DI PALERMO

Progetto Tecniche per la gestione degli Open-Data:

A cura di:
Benedetto Marino 0643961
Massimiliano Poma 0640469

Sommario

1 IDEA DI PROGETTO 2

1.1 Project idea 2

2 DATASET UTILIZZATI E LICENZE 3

3 PIPELINE DI ELABORAZIONE 6

3.1 Filtraggio dataset 8
3.2 Standardizzazione dataset 8
3.3 Calcolo coordinate 8
3.4 Calcolo indice vivibilità 9
3.5 Ontologia 9
3.6 Creazione URI 10
3.7 Interlinking e TTL 11

4 VISUALIZZAZIONE 12

1. Idea di progetto

CityRank nasce dall'idea di stilare una classifica riguardante il grado di vivibilità delle cinque province italiane più popolose, ovvero Palermo, Napoli, Roma, Milano e Torino.
L'indice viene calcolato prendendo in considerazione alcuni fattori che incidono nella quotidianità cittadina, ad esempio il reddito medio, la raccolta differenziata o l'indice di criminalità. I dati finali sono visualizzati su una mappa in cui vengono indicate le province con i rispettivi indici e strutture, quali scuole, ospedali e biblioteche.

1.1 Project Idea

CityRank was born from idea of draw up a ranking about the livability of the five most populous Italian provinces, that are Palermo, Napoli, Roma, Milano and Torino.
The index is calculated considering some factors that affect everyday life in the city, for example average income, recycling or the crime rate. The final data are displayed on a map showing the provinces with their respective index and informations such as schools, hospitals and libraries.

2. Dataset utilizzati e licenze

Per la realizzazione del progetto sono stai utilizzati 21 dataset, prelevati dalle banche dati dell'Istat, dell'Inps o da quelle dei siti ministeriali. Qui un elenco con le rispettive fonti e le licenze.

- **Redditi_per_fascia_di_reddito_su_base_comunale.csv:** Contiene informazioni sui redditi e principali variabili Irpef su base comunale. Viene distribuito con licenza CC-BY 3.0 al link https://www1.finanze.gov.it/.
- **Retribuzione_media.csv:** Contiene informazioni riguardanti la retribuzione media nell'intero territorio nazionale. Viene distribuito con licenza CC-BY 3.0 al link Statistiche Istat.
- **Torino_milano_inps.csv:** Contiene informazioni riguardanti le pensioni di vecchiaia erogate nelle province di Torino Milano. Viene rilasciato con licenza IODL2.0 al link https://www.inps.it.
- **Napoli_inps.csv:** Contiene informazioni riguardanti le pensioni di vecchiaia erogate nelle provincia di Napoli. Viene rilasciato con licenza IODL2.0 al link https://www.inps.it.
- **Palermo_inps.csv:** Contiene informazioni riguardanti le pensioni di vecchiaia erogate nelle provincia di Palermo. Viene rilasciato con licenza IODL2.0 al link https://www.inps.it.
- **Roma_inps.csv:** Contiene informazioni riguardanti le pensioni di vecchiaia erogate nelle provincia di Roma. Viene rilasciato con licenza IODL2.0 al link https://www.inps.it.
- **Rifiuti_urbani_2018.csv:** Contine informazioni riguardanti la produzione di rifiuti pro capite e la percentuale di rifiuti differenziati su base provinciale. Viene rilasciato con licenza CC-BY 3.0 al link Statistiche Istat.

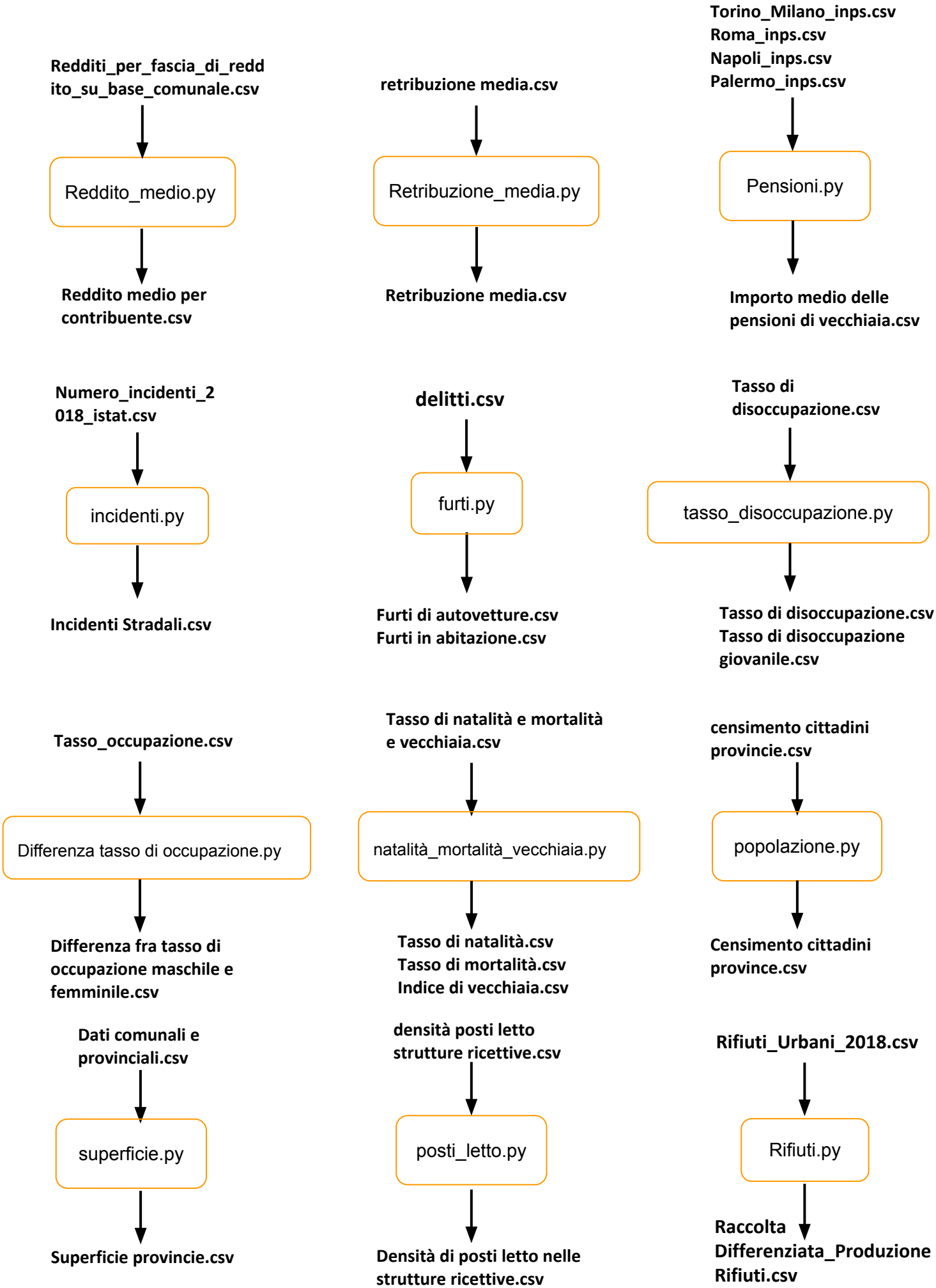
- **Numero_incidenti_2018_istat.csv:** Contiene informazioni riguardanti il numero di incidenti stradali, con annessi morti e/o feriti su base provinciale. Viene rilasciato con licenza CC-BY 3.0 al link Statistiche Istat.
- **delitti.csv:** Contiene informazioni riguardanti tutti i tipi di reati denunciati dalle forze dell'ordine su base provinciale. Viene rilasciato con licenza CC-BY 3.0 al link Statistiche Istat.
- **Tasso di disoccupazione.csv**: Contiene informazioni riguardanti il tasso di disoccupazione, diviso per classi d'età, relativo alle province nazionali. Viene rilasciato con licenza CC-BY 3.0 al link Statistiche Istat.
- **Tasso_occupazione.csv:** Contiene informazioni riguardanti il tasso di occupazione, diviso per classi d'età, relativo alle province nazionali. Viene rilasciato con licenza CC-BY 3.0 al link Statistiche Istat.
- **Tasso_natalità_mortalità_vecchiaia.csv**: Contiene informazioni riguardanti i tassi di natalità, mortalità e vecchiaia divisi per provincia. Viene rilasciato con licenza CC-BY 3.0 al link Statistiche Istat.
- **Censimento cittadini province.csv:** Contiene informazioni riguardanti il numero di abitanti divisi per province. Viene rilasciato con licenza CC-BY 3.0 al link Statistiche Istat.
- **Densità posti letto strutture ricettive.csv:** Contiene informazioni riguardanti il numero dei posti letto nelle strutture ricettive, il tipo di struttura, il numero totale delle strutture su base comunale. Viene rilasciato con licenza CC-BY 3.0 al link Statistiche Istat.
- **AGCOM_BBmap_opendata_province.csv:** Contiene informazioni riguardanti il numero di abbonamenti per la connessione a internet e il tipo di connessione su base provinciale. Viene rilasciato con licenza CC_BY 4.0 al link https://maps.agcom.it/.
- **Dati comunali e provinciali.csv:** Contiene informazioni riguardanti la superficie delle province espressa in ettari o in km^2 e il numero degli abitanti. Viene rilasciato con licenza CC-BY 3.0 al link Statistiche Istat.
- **Ospedali_italiani.csv:** Contiene informazioni riguardanti le strutture sanitarie presenti sull'intero territorio nazionale. Viene rilasciato con licenza IODL2.0 al link http://www.dati.salute.gov.it.
- **SCUANAGRAFESTAT20192020190901.csv:** Contiene informazioni relative all'anagrafe delle scuole presenti sull'intero territorio nazionale. Viene rilasciato con licenza IODL2.0 al link https://dati.istruzione.it.

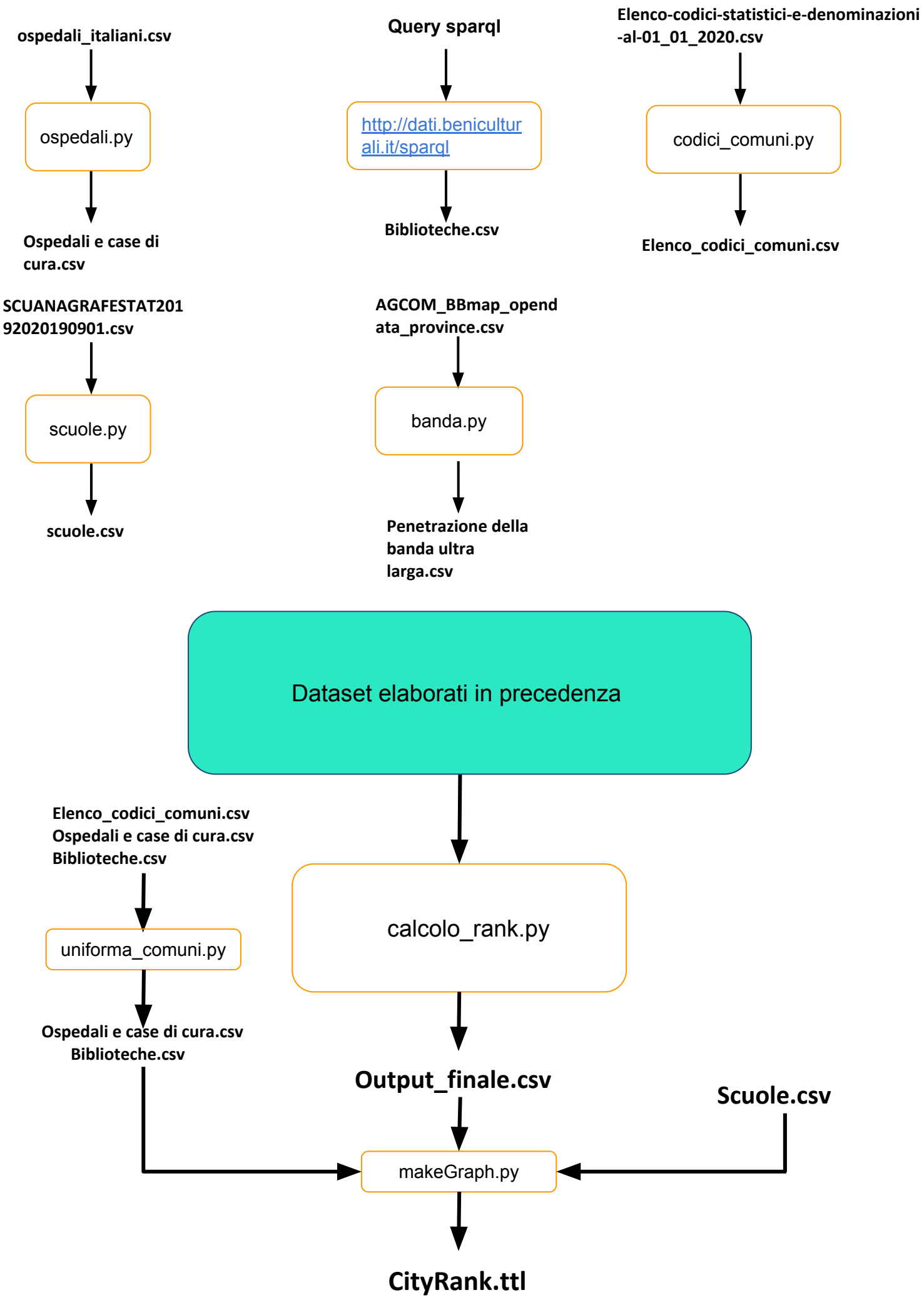
- **Elenco-codici-statistici-e-denominazioni-al-01_01_2020.csv:** Contiene l'elenco di tutti i comuni italiani con relativo codice catastale.Viene rilasciato con licenza CC-BY 3.0 al link Statistiche Istat.
- **Biblioteche.csv:** Contiene le informazioni riguardanti le biblioteche presenti nelle cinque province prese in esame nel progetto. I dati sono stati estratti interrogando l'endpoint sparql del ministero dei beni culturali. I dati vengono rilasciati con licenza CC-BY 3.0.

Query utilizzata per estrarre i dati delle biblioteche.

```
SELECT distinct ?Provincia ?Biblioteca ?URI ?Indirizzo ?CAP ?Latitudine ?Longitudine
WHERE {
      ?URI a cis:Library; cis:hasSite ?site.
      ?site cis:siteAddress ?via.
      ?via clvapit:hasProvince ?p.
      ?p l0:name ?Provincia.
      ?URI cis:institutionalCISName ?Biblioteca.
      ?via clvapit:fullAddress ?Indirizzo; clvapit:postCode ?CAP.
      ?site clvapit:hasGeometry ?geo.
      ?geo clvapit:lat ?Latitudine; clvapit:long ?Longitudine.
      ?site cis:siteAddress ?adress.
      ?adress clvapit:hasProvince ?provincia.
      FILTER(?provincia IN( <http://dati.beniculturali.it/iccu/anagrafe/resource/Province/201>,
                            <http://dati.beniculturali.it/iccu/anagrafe/resource/Province/215>,
                            <http://dati.beniculturali.it/iccu/anagrafe/resource/Province/258>,
                            <http://dati.beniculturali.it/iccu/anagrafe/resource/Province/263>,
                            <http://dati.beniculturali.it/iccu/anagrafe/resource/Province/282>))
}
```

3. Pipeline di elaborazione




ospedali_italiani.csv
ospedali.py
Ospedali e case di cura.csv
Query sparql
http://dati.beniculturali.it/sparql
Biblioteche.csv
Elenco-codici-statistici-e-denominazioni-al-01_01_2020.csv
codici_comuni.py
Elenco_codici_comuni.csv
SCUANAGRAFESTAT20192020190901.csv
scuole.py
scuole.csv
AGCOM_BBmap_opendata_province.csv
banda.py
Penetrazione della banda ultra larga.csv
Dataset elaborati in precedenza
Elenco_codici_comuni.csv
Ospedali e case di cura.csv
Biblioteche.csv
uniforma_comuni.py
Ospedali e case di cura.csv
Biblioteche.csv
calcolo_rank.py
Output_finale.csv
Scuole.csv
makeGraph.py
CityRank.ttl

3.1 Filtraggio dataset

I dataset iniziali contenevano le informazioni a livello nazionale, da questi sono quindi state estratte solamente le informazioni relative alle province di Palermo, Napoli, Roma, Milano e Torino.
Il processo di filtraggio è stato effettuato principalmente tramite l'utilizzo di due librerie per python, ovvero pandas e csv.

3.2 Standardizzazione dataset

Una volta filtrati tutti i dataset vi era la necessità di rendere le informazioni tra questi il più uniformi possibili, in modo da facilitare le elaborazioni successive. In particolare è stato necessario andare ad aggiungere i codici catastali dei comuni nei dataset degli ospedali e delle biblioteche.
Per questo scopo è stato utile scaricare dalla sezione open data dell'ISTAT un dataset contenente i comuni italiani e i rispettivi codici catastali.
Quest'ultimo dataset è stato poi utilizzato come base nello script uniforma_comuni.py, nel quale si ricercano i comuni presenti nei dataset delle biblioteche e degli ospedali e quindi si ricava il codice catastale corrispondente.

3.3 Calcolo coordinate

Una volta filtrati ed uniformati i dataset si dovevano trovare le coordinate delle scuole e degli ospedali, così da poter poi creare una mappa.
Per il calcolo delle coordinate delle strutture è stato utilizzato Nominatim, un servizio gratuito messo a disposizione da **OpenStreetMap**(OSM).
Prima di interrogare Nominatim è stata effettuata una "pulizia" delle informazione, la quale consiste nell'eliminare le abbreviazioni del tipo "P.zza", "V.le","C.so" ecc ecc...
A questo punto sono state utilizzate le API di openstreetmap tramite la libreria geopy. Nel caso in cui OSM non restituisse le coordinate di una struttura si cercava di ottenere quelle del comune di appartenenza.
Per tutte le strutture per le quali non sono state ricavate nemmeno le coordinate del comune è stato assegnato (0,0).

3.4 Calcolo indice vivibilità

Il calcolo dell'indice viene effettuato in calcolo_rank.py, in particolare viene creata una classifica parziale per ogni indicatore, assegnado un punteggio di 0.5pt alla prima classificata, 0.1pt all'ultima classificata.
La classifica finale è stata quindi calcolata effettuando la somma dei punteggi ottenuti in base alle varie classifiche parziali.
L'output di questa fase è Output_finale.csv, nel quale sono presenti le classifiche per ciascun indicatore, la classifica finale e le coordinate delle province.

3.5 Ontologia

L'ontologia è stata creata con l'ausilio di Protégé. Sono state definite cinque classi :

- Provincia
- Comune
- Scuola
- Ospedali
- Biblioteche

Inoltre sono state utilizzate tre classi dell'ontologia di schema.org, in particolare:

- GeoCoordinates
- Place (Usata come superclasse delle cinque classi da noi create)
- PostalAddrees

Place è stato utilizzato come superclasse di Provincia, Comune, Scuola, Ospedali e Biblioteche. Poi sono state definite tre object properties:

- haIstituto
- haProvincia
- intotolatoA

Anche qui sono state utilizzate due object properties di schema.org e una di W3C cioè:

- owl.sameAs
- Place.geo
- GeoCoordinates.address

Infine sono state definite trenta data properties.

3.6 Creazione URI

Per la creazione delle URI delle nostre risorse sono state attuate diverse strategie.

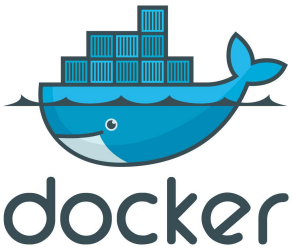
- **Province:** Per ogni provincia è stata creata una URI del tipo *http://www.cityrank.org/resource/NomeProvincia*.
- **Comuni:** Per le URI del comune è stato utilizzato il codice catastale del comune. Ad esempio l'URI del comune di palermo è *http://www.cityrank.org/resource/G273*.
- **Scuole:** Per le URI delle scuole è stato utilizzato il codice della scuola, presente nel dataset iniziale. L'URI sarà del tipo *http://www.cityrank.org/resource/miaa80901p*.
- **Biblioteche:** Per le URI delle biblioteche è stato utilizzato l'identificativo usato anche nell'ontlogia del MIBAC. L'URI sarà del tipo *http://www.cityrank.org/resource/IT-MI0001*.
- **Ospedali:** Per gli ospedali non era presente un codice univoco, per rimediare è stata usata una funzione di hashing che calcolasse un digest esadecimale tramite l'algoritmo ripmed160. Se viene calcolato un digest già presente, viene concatenato un "2" a quest'ultimo fino a quando non si ottiene un nuovo digest. L'URI degli ospedali sarà quindi del tipo *http://www.cityrank.org/resource/00fe60fcba3ea596a79f812c196e5a0c8d00c743*.

Oltre alle URI delle istanze delle classi da noi definite, vi era la necessità di creare una URI per gli oggetti PostalAddress e GeoCoordinates. Per fare ciò è sono stati utilizzati i codici delle nostre risorse preceduti da "loc-" nel caso di un oggetto GeoCoordinates, "addr-" in caso di un oggetto di tipo PostalAddress.
Es:
Volendo identificare le coordinate e l'indirizzo per la risorsa *http://www.cityrank.org/resource/scuole/miaa80901p* avremo

- **GeoCoordinates:** *http://www.cityrank.org/resource/scuole/loc-miaa80901p*
- **PostalAddress:** *http://www.cityrank.org/resource/scuole/addr-miaa80901p*

3.7 Interlinking e file ttl

L'interlinking dei dati è stato fatto con DBpedia e il MIBAC (Ministero per I Beni e le Attività Culturali), per le province ed i comuni è stata costruita l'URI della risorsa e successivamente si effettua una richiesta a DBpedia, se la risorsa esiste la associamo alla nostra tramite la proprietà sameAs di owl.
Per l'interlinkign delle biblioteche è bastato associare l'URI prelevata con query sparql a quella della nostra risorsa tramite la proprietà sameAs.
Un'altra strategia di interlinking è stata quella di individuare la persona alla quale era intitolata una scuola o un ospedale. Per fare ciò si è utilizzato DBpedia spotlight, un servizio di Entity Recognition. In particolare è stata prima scaricata l'immagine di DBpedia spotlight da docker.hub e quindi avviata in locale, a questo punto è bastato passare i nomi delle scuole e degli ospedali tramite le API di DBpedia spotlight, ottenendo quindi l'URI della risorsa individuata.



Sia la creazione delle URI che l'Interlinking avvengono in makeGraph.py che prende in input Output_finale.csv, il dataset delle biblioteche, degli ospedali e delle scuole con l'aggiunta dei codici dei comuni. All'interno dello script python si associa ogni risorsa contenuta nei dataset alla corrispondente classe della nostra ontologia, specificando poi le object properties e le data properties.

4. Visualizzazione

Per la visualizzazione dei dati elaborati abbiamo deciso di utilizzare un altro servizio che si basa su OSM, cioè Umap. All'interno della mappa sono stati inseriti quattro diversi livelli :

- Il primo livello (Marcatore blu) contiene le province e quando l'utente clicca su di esso compaiono tutti gli indicatori e l'indice finale.

- Il secondo livello (Marcatore rosso) contiene le informazioni relative agli ospedali, quindi: nome struttura, il tipo, l'indirizzo ecc.

- Il terzo livello (Marcatore giallo) contiene le informazioni delle scuole. Per questo dataset abbiamo deciso di eseguire una pre-elaborazione prima di visualizzarlo con Umap. In molti casi le coordinate erano pari a zero o nominatim restituiva delle coordinate fuori dal territorio italiano. Per risolvere il problema abbiamo eseguito lo script scuoleToUmap.py che prende in input il dataset delle scuole e elimina tutte quelle strutture che hanno coordinate pari a zero o coordinate che non rientrano in un bounding box che include tutto il territorio italiano.

- Il quarto livello (Marcatore verde) contiene le informazioni delle biblioteche.

La mappa completa è visualizzabile al link http://u.osmfr.org/m/477407/

INDICATORI

Provincia di Roma
Reddito medio complessivo per contribuente: 24835.85
Retribuzione media: 15.42
Importo medio delle pensioni di vecchiaia: 1562.38
Produzioni di rifiuti urbani pro capite: 603.4 Kg/Abitante
Raccolta differenziata: 42.9%
Rapporto Superficie/Numero Ospedali: 42.57
Rapporto Superficie/Numero Scuole: 2.23
Incidenti stradali: 471.46
Furti di autovetture: 398.07
Furti in abitazione: 273.62
Tasso di disoccupazione: 9.1%
Tasso di disoccupazione giovanile: 21.45%
Differenza tra tasso di occupazione maschile e femminile: 12.92%
Tasso di natalità: 7.2
Tasso di mortalità: 9.5
Indice di vecchiaia: 157.2
Numero posti letto nelle strutture ricettive: 292829
Numero biblioteche ogni 10000 abitanti: 2.92
Penetrazione della banda ultra larga: 33.64

Indice di vivibilità: 6.0