ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA

CAMPUS DI CESENA

SCUOLA DI SCIENZE

CORSO DI LAUREA TRIENNALE IN INGEGNERIA E SCIENZE INFORMATICHE

PROGETTAZIONE DI UNA PIATTAFORMA PER L'ANALISI DEI DATI DI TRAIETTORIA:

UN CASO DI STUDIO SUI DATI DI NAVIGAZIONE

Tesi in

Laboratorio di Basi di Dati

Relatore:
Prof. Matteo Golfarelli

Presentata da:
Chiara Forresi

Correlatore:
Dott. Matteo Francia

Sessione I
Anno Accademico 2017/2018

Alla mia famiglia, ai miei amici,
a tutti coloro che mi sono stati vicini
e mi hanno supportato dall'inizio alla fine.

Introduzione

Negli ultimi anni si è assistito ad una crescente disponibilità di dati di traiettoria, ovvero dati relativi agli spostamenti di oggetti di diverso tipo, provenienti da innumerevoli fonti quali smartphone e dispositivi di sensoristica varia. Questo processo è stato sicuramente favorito dalla continua crescita, sia in termine tecnologico, sia di utilizzo, di tali dispositivi. Ciò ha portato da un lato a una sempre maggiore precisione ed accuratezza dei dati e, dall'altro, a un aumento consistente delle quantità degli stessi. A partire da questi presupposti l'interesse si è indirizzato verso la possibilità di ricavare conoscenza dai dati grezzi, implementando uno stack di algoritmi tali da estrarre informazioni significative. Questi algoritmi includono la scoperta degli **stay point** (punti di permanenza dell'utente) e la costruzione delle traiettorie, che combinati possono essere utili per ricavare pattern frequenti degli oggetti nel dominio applicativo.
Le possibilità sono molteplici, vista la varietà dei dati in gioco, è infatti possibile studiare i comportamenti: delle persone (singole o in gruppo), dei veicoli di trasporto, degli animali e dei fenomeni naturali. L'interpretazione dei dati varia, dunque, in base al dominio applicativo e dalla specifica conoscenza che si vuole estrarre.

In questo lavoro di tesi si è progettato e sviluppato un insieme di algoritmi atti ad analizzare dati di traiettoria navali. L'algoritmo per l'analisi delle traiettorie si occupa di estrarre le traiettorie delle imbarcazioni basandosi sui dati grezzi, precedentemente filtrati. Successivamente, attraverso i dati estratti, è stato possibile analizzare i risultati e ricavare informazioni riassuntive da rapportare ai dati che ci si aspetta di avere nel mondo reale in questo

specifico ambito. L'algoritmo, di per se applicabile a qualsiasi contesto, è stato specializzato nel dominio della navigazione commerciale e in particolare su un caso di studio dei dati delle imbarcazioni che hanno navigato nel territorio degli Stati Uniti nei primi tre mesi del 2014. Questi dati sono stati arricchiti con dati open relativi ai porti, ai confini e alle zone navigabili, di commercio e corsie del territorio. Tutti i dati ottenuti dalle varie elaborazioni sono visualizzabili graficamente nella piattaforma web sviluppata.

I risultati ottenuti hanno evidenziato che i dati di partenza sono abbastanza coerenti con la realtà: sebbene la sparisità degli stessi all'interno delle categorie di imbarcazione risulti notevole, la verità di fondo risulta rispettata. Inoltre e soprattutto, ci si è resi conto che è possibile generalizzare gli algoritmi e la piattaforma sviluppata nei vari domini applicativi possibili, avendo cura di regolare i parametri in base al contesto a cui si vogliono applicare. Infine, si è cercato di essere più indipendenti possibili dalla piattaforma utilizzata per memorizzare i dati quindi, sebbene questo progetto usi un database Oracle, la migrazione verso altre piattaforme, sia di tipo relazionale, sia big data (data la quantità dei dati), risulta pressoché immediata.

Il lavoro di tesi è organizzato nei seguenti capitoli:

1. **analisi dei dati di traiettoria** dove vengono introdotti termini e concetti usati nell'ambito delle informazioni generate dagli spostamenti;
2. **le tecnologie utilizzate** in cui vengono descritte le tecnologie utilizzate per implementare lo stack algoritmico;
3. **la piattaforma** in cui viene descritta l'architettura della piattaforma sviluppata;
4. **tecniche per l'analisi dei dati** dove vengono descritte le tecniche impiegate per analizzare i dati e i pattern frequenti;
5. **il prototipo realizzato** in cui viene presentato il prototipo realizzato e i relativi risultati ottenuti.

Indice

Elenco delle figure

Capitolo 1

Analisi dati di traiettoria

Per addentrarci all'interno del contesto della tesi, in questo capitolo vengono introdotti alcuni termini e concetti usati nell'ambito delle informazioni generate dagli spostamenti.

1.1 Traiettoria

Una **traiettoria** descrive gli spostamenti di un oggetto, può essere dunque definita come *una traccia generata da un oggetto in movimento in spazi geografici, solitamente rappresentata da una serie di punti ordinati cronologicamente* [15]. Negli ultimi anni si è assistito ad un vero e proprio sviluppo tecnologico in ambito di acquisizione delle posizioni che hanno generato grandi quantità di dati spaziali, dai quali è possibile estrarre traiettorie, relativi alla mobilità di oggetti, ad esempio di persone, veicoli o animali. A partire da questi dati si è, quindi, pensato di estrarre delle informazioni utili per capire lo spostamento degli oggetti e la loro posizione, promuovendo un'ampia gamma di applicazioni nei social network basati sulla posizione [14]. Ciò è alla base del *trajectory data mining*. Come già detto una traiettoria può essere relativa a contesti differenti i quali, come descritto da Zheng in [15], possono essere classificati in:

1. mobilità dei veicoli di trasporto: quotidianamente ci circondano un elevato numero di veicoli dotati di *GPS* (sistema di posizionamento globale), come taxi, bus, navi e aerei, i quali in continuazione generano dati composti almeno dal riferimento spaziale del punto in cui si trovano e un *timestamp* (marca temporale). Ordinando cronologicamente questi ultimi sarà, quindi, possibile costruire traiettorie che potranno essere usate, ad esempio per: assegnazione delle risorse, analisi del traffico e migliorare la rete di trasporti.

2. mobilità delle persone: i movimenti delle persone vengono raccolti per lunghi periodi, in modo attivo e passivo. Le informazioni vengono raccolte in modo attivo quando l'individuo, ad esempio, vuole tener traccia dei suoi viaggi attraverso il GPS e condividerli con i suo amici, oppure vuole raccogliere informazioni relative ad un'attività sportiva. In caso di raccolta passiva dei dati, l'utente, invece, genera dati spaziali involontariamente attraverso, ad esempio, la sua rilevazione da parte di ripetitori o lo svolgimento di transazioni mediante bancomat.

3. mobilità degli animali: la memorizzazione di traiettorie di animali, come tigri e uccelli, può essere utile ai biologi per monitorare i fenomeni di migrazione, i comportamenti e le condizioni di vita.

4. mobilità dei fenomeni naturali: meteorologi, ambientalisti, climatologi e oceanografi raccolgono dati di traiettoria relativi ad alcuni fenomeni naturali come: uragani, tornadi e correnti oceaniche. Questi dati catturano i cambiamenti ambientali e climatici, aiutando gli scienziati ad affrontare disastri naturali e progettare l'ambiente naturale dove viviamo.

1.1.1 Costruzione di traiettorie

Come accennato in precedenza, le varie tecnologie (come il GPS) ci fornisco i punti grezzi, non le traiettorie. Tali punti sono solitamente caratterizzati

da una serie di informazioni che dipendono dal contesto in cui ci troviamo, ma necessariamente hanno un riferimento spaziale del punto in cui si trova l'oggetto e il relativo timestamp. Dunque, considerando la definizione di cui sopra, basterebbe ordinare cronologicamente i punti per ogni singolo oggetto appartenente al contesto in esame ed estrarre traiettorie da questo set di dati. Ma c'è un problema: i punti raccolti potrebbero essere molti, più dei necessari (dipendentemente dai contesti applicativi), e soprattutto potrebbero contenere del rumore. Tenendo in considerazione queste problematiche è, perciò, necessaria una pre-elaborazione dei dati. Questo processo può essere sintetizzato in quattro fasi [15]:

1. filtraggio del rumore;
2. scoperta degli *stay point* (punti di permanenza dell'utente);
3. compressione delle traiettorie;
4. segmentazione delle traiettorie.

Filtraggio del rumore

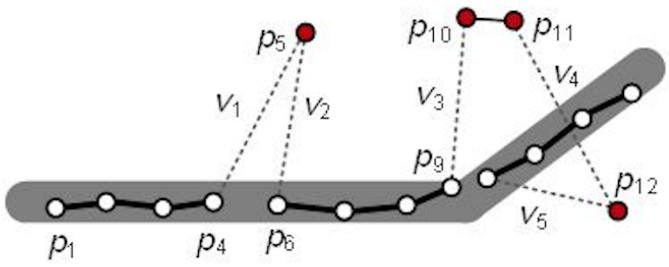


Figura 1.1: Esempio di una traiettoria con rumore.

Adattata da:

https://www.microsoft.com/en-us/research/wp-content/uploads/2016/03/trajectorycomputing-preprocessing.png

I dati spaziali non sempre sono accurati, come mostrato in Figura 1.1. I rumori dei sensori o i cali di segnale possono procurare errori a volte accetta-

bili e rimediabili attraverso particolari algoritmi per trovare la corrispondenza nella mappa, altre volte davvero grandi e dai quali non è possibile estrarre informazioni utili. Questo processo può essere svolto attraverso diverse tecniche, una fra le più efficienti è basata su un'euristica e l'uso di un algoritmo per rimuovere le anomalie. Tutto ciò si incentra sulla considerazione che il numero di punti con rumore è probabilmente inferiore al numero di punti totale. Sostanzialmente vengono calcolate le differenze di velocità tra le varie coppie di punti (ordinati cronologicamente) e il segmento con una velocità maggiore di una certa soglia verrà tagliato fuori e considerato come un'anomalia. Nell'esempio in Figura 1.1 i segmenti tratteggiati vengono identificati come rumore.

Scoperta degli stay point

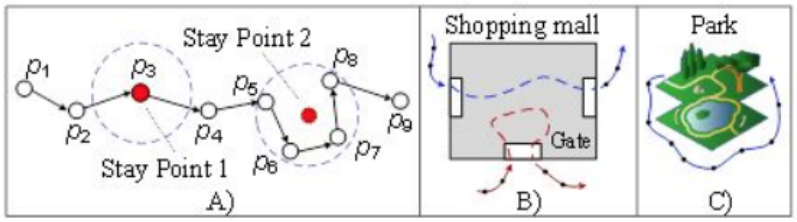


Figura 1.2: Esempio di stay point in una traiettoria.

Adattata da: http://slideplayer.com/9360561/28/images/11/Stay+Point+Detection+Some+points+denote+locations+where+people+have+stayed+for+a+while.+Shopping+malls+and+tourist+attractions..jpg

I punti che compongono una traiettoria non avranno tutti la stessa importanza: ci saranno punti in cui l'utente rimarrà per poco tempo, altri in cui, invece, rimarrà fermo per del tempo consistente come lo *Stay point 1* in Figura 1.2 e altri ancora in cui si muoverà attorno per del tempo (ad esempio un centro commerciale Figura 1.2(b) o un parco Figura 1.2(c)) come lo *Stay point 2* in Figura 1.2 [15]. Il primo algoritmo proposto per questo problema, consiste prima nel controllare se la distanza tra "punto ancora" ed i suoi successori è in una traiettoria maggiore di un certo parametro (es. 100 m). L'algoritmo misura la differenza di tempo tra il punto ancora e il suo ultimo

successore, se è maggiore di un certo parametro viene scoperto uno stay point e l'algoritmo riparte alla ricerca di un presunto stay point successivo. Tale implementazione è stata ottimizzata attraverso l'uso del clustering di densità (del quale parleremo nel Capitolo 4).

Compressione delle traiettorie

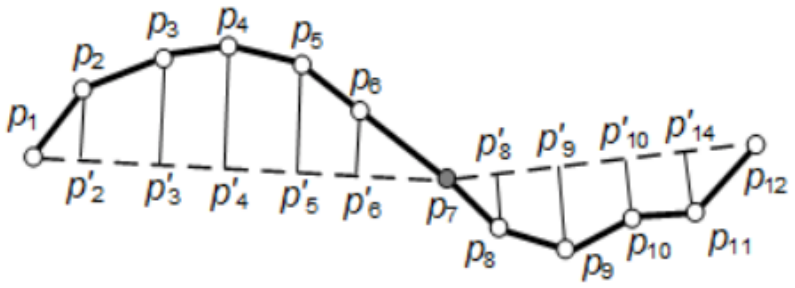


Figura 1.3: Esempio di compressione di una traiettoria.

Adattata da:

https://www.microsoft.com/en-us/research/wp-content/uploads/2016/03/trajectorycomputing-preprocessing.png

Una traiettoria potrebbe essere formata da un elevato numero di punti e dunque può essere utile una compressione della stessa, come nell'esempio in Figura 1.3. Esistono due categorie principali di tecniche di compressione di traiettorie: *compressione offline* e *compressione online*. La prima riduce la dimensione di una traiettoria dopo la sua generazione, la seconda, invece, fa si che la compressione avvenga istantaneamente (mentre vengono ricevuti nuovi punti). Nel dettaglio la compressione offline ha come obiettivo quello di generare una traiettoria approssimata, scartando dei punti con un errore trascurabile dalla traiettoria originaria. Uno degli algoritmi più conosciuti per questo problema è quello di Douglas-Peucker [12], il quale obiettivo è dunque di riuscire ad approssimare con un segmento di linea la traiettoria di partenza. Esso, ricorsivamente, suddivide il problema in due selezionando come punti di divisione quei punti che contribuiscono al maggiore errore.

Questo processo continua finché l'errore tra la traiettoria originaria e la nuova è sotto una specifica soglia.
Poiché esistono applicazioni che richiedono di trasmettere le traiettorie realtime, in questi casi questo approccio non risulta esauriente e, quindi, si passa alla *compressione online*. La quale decide sul momento se un nuovo punto acquisito dovrebbe essere o meno conservato in una traiettoria. Esistono due principali categorie con questo approccio: una è basata sulle finestre e l'altra sulla velocità e la direzione degli oggetti in movimento. Il primo approccio ha, a sua volta, due algoritmi fondamentali. Uno è lo *Sliding Window* (finestra scorrevole) la cui idea è quella di adattare i punti in una finestra scorrevole che continua a crescere finché non viene superata una certa soglia di errore. L'altro, *Open Window* (finestra aperta), usa l'euristica dell'algoritmo Douglas-Peucker, quindi per approssimare la traiettoria sceglie il punto che nella finestra ha l'errore maggiore. Questo punto sarà poi usato come ancora per approssimare i suoi successori. Nell'altra categoria di compressione online, che invece si basa sulla velocità e la direzione degli oggetti in movimento, troviamo l' algoritmo di Potamias [10]. Il quale usa un'area protetta basata sulle ultime due posizioni rilevate e una data soglia per determinare se un nuovo punto acquisito contiene informazioni importanti. Se questo punto si trova nei limiti dell'area stabilita è considerato ridondante e può essere scartato, altrimenti viene incluso nella nuova traiettoria.
Esiste, un'altra categoria di tecniche di compressione che si basa sul significato semantico dei punti, infatti, ci sono dei punti dove gli utenti stanno per più tempo o dove cambiano molto frequentemente direzione. Questi punti potrebbero avere maggior significato rispetto agli altri. Un algoritmo di questa categoria è di Chen [2], si basa sulla divisione della traiettoria in segmenti in cui l'utente passeggia o non passeggia e un punto ha un peso basato sul grado di variazione di direzione e sulla distanza dai vicini.
Un altro ramo di ricerca considera la compressione delle traiettorie con i vincoli presenti nella rete di trasporti. Ad esempio, un punto può essere trascurato se il movimento continua sul percorso più breve dal punto definito di

ancoraggio nella posizione corrente. Questo approccio solitamente necessita di algoritmi di corrispondenza con la mappa.

Segmentazione delle traiettorie

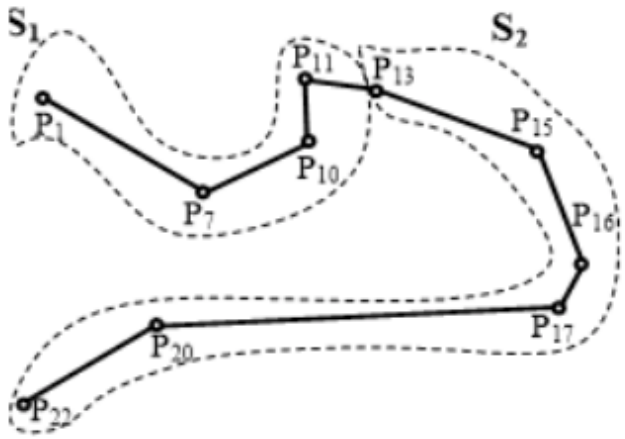


Figura 1.4: Esempio di segmentazione di in traiettoria.

Adattata da:

https://www.microsoft.com/en-us/research/wp-content/uploads/2016/03/trajectorycomputing-preprocessing.png

In alcuni casi può essere utile segmentare una traiettoria al fine di estrarre conoscenza, ad esempio pattern dai singoli segmenti. Questo processo porta anche alla riduzione della complessità della traiettoria stessa. Un esempio di segmentazione è in Figura 1.4. In generale una traiettoria può essere compressa in tre modi:

- intervallo di tempo: ogni qualvolta che si supera un certo intervallo di tempo la traiettoria viene spezzata e viene, dunque, definita una sotto-traiettoria;
- forma della traiettoria: il parametro di suddivisione in questo caso è il cambiamento della direzione;
- significato semantico: i segmenti si creano basandosi su un significato semantico particolare, ad esempio sugli stay point contenuti in una

traiettoria. Un altro esempio potrebbe essere dividere le traiettorie in base al mezzo utilizzato (a piedi, bici, macchina) o in traiettorie in movimento e ferme, quindi attenendosi alla velocità degli oggetti.

1.2 Personal gazetteer e Stay Point

Un *personal gazetteer* (dizionario geografico personale) *registra i luoghi significativi di una specifica persona* [17]. Un personal gazetteer può contenere, ad esempio, per ciascun luogo un'etichetta e una sua rappresentazione geometrica (come un punto, un gruppo di punti, un'area, etc.).
Dunque, come costuire un personal gazeteer?

Già nella sezione precedente è stato introdotto il concetto di stay point. Un personal gazetter non è altro che l'insieme degli stay point di un determinato utente. Nello specifico uno stay point indica una regione geografica in cui un utente è rimasto in un determinato intervallo di tempo. L'estrazione degli stay point dipende da due parametri di scala, una soglia di tempo (T_{threh}) e una di distanza (D_{threh}). Pertanto, come nei punti $\{p_5, p_6, p_7, p_8\}$ in figura 1.2, un singolo stay point può essere considerato come una posizione virtuale caratterizzata da un gruppo di punti GPS consecutivi P=$\{p_{\text{m}}, p_{\text{m+1}}, \ldots, p_{\text{n}}\}$, dove $\forall$ m $<$ i $\leq$ n, Distanza$(p_{\text{m}}, p_{\text{i}}) \leq D_{\text{threh}}$ e $|p_{\text{n}} * T - p_{\text{m}} * T| \geq T_{\text{threh}}$. Formalmente uno stay point, condizionato da P, D_{threh} e T_{threh}, può essere definito come s=(Lat, Lngt, arvT, levT), dove:

1. $s.Lat = \sum_{i=m}^{n} p_i * Lat/|P|$
2. $s.Lngt = \sum_{i=m}^{n} p_i * Lngt/|P|$

indicano rispettivamente latitudine e longitudine media della collezione P, e s.arvT= p_{m}.T e s.levT = p_{n}.T rappresentano l'orario di arrivo e di partenza di un utente su s [16].

Capitolo 2

Le tecnologie utilizzate

In questo capitolo è descritta la base tecnologica su cui si è basato il progetto di tesi.

2.1 Oracle Spatial

Il DBMS Oracle è dotato di uno specifico componente per la gestione in modo rapido ed efficiente dei dati spaziali: *Oracle Spatial and Graph* [6]. Quest'ultimo consiste in un insieme integrato di funzioni, procedure, tipi e modelli di dati a supporto dei processi di archiviazione, accesso e analisi dei dati spaziali. Una volta che i dati spaziali sono archiviati in un database Oracle, possono essere facilmente manipolati, recuperati e correlati a tutti gli altri dati memorizzati nel database.

Le funzionalità spaziali forniscono uno schema e funzioni che facilitano l'archiviazione, il recupero, l'aggiornamento e l'interrogazione sui dati spaziali in un database Oracle. Le principali caratteristiche di Spatial e Graph sono:

- uno schema (MDSYS) che prescrive l'archiviazione, la sintassi e la semantica dei tipi di dati geometrici supportati;
- un meccanismo di indicizzazione spaziale;

- operatori, funzioni e procedure per l'esecuzione di query di area di interesse e unione spaziale e altre operazioni di analisi spaziale;
- funzioni e procedure per le operazioni di utility e tuning;
- modello di dati topologici per lavorare con i dati su nodi, spigoli e facce di una topologia;
- modello di dati di networking per la rappresentazione di funzionalità o oggetti modellati come nodi e collegamenti (vertici e spigoli) in un grafico;
- GeoRaster, una funzionalità che consente di archiviare, indicizzare, interrogare, analizzare e distribuire dati GeoRaster, ovvero immagini raster e dati sotto forma di griglia e i relativi metadati associati.

2.2 I dati spaziali

Un esempio comune di dati spaziali può essere una cartina geografica, essa si può percepire come un oggetto bidimensionale che contiene punti, linee e poligoni che possono rappresentare città, strade e confini politici come stati o province. Si tratta, dunque, di una visualizzazione di informazioni geografiche in cui le posizioni dei vari oggetti sulla superficie della Terra vengono proiettate su un display bidimensionale o su un pezzo di carta, preservando le posizioni e le distanze relative degli oggetti sottoposti a rendering.

I dati che indicano le posizioni di questi oggetti rispetto alla Terra, come longitudine e latitudine, sono i dati spaziali. Un *GIS* (Geographic Information System) viene spesso utilizzato per archiviare, recuperare e renderizzare questi dati spaziali relativi alla Terra.

La componente spaziale di una *feature spaziale* è la rappresentazione geometrica della sua forma in uno spazio di coordinate, come indicato nella sua geometria [7]. Quest'ultima è memorizzata nel tipo di dati spaziali nativi di Oracle per dati vettoriali, *SDO_GEOMETRY*, basato sul modello relazionale ad oggetti.

2.2.1 Geometrie

Il modello di dati spaziali è una struttura gerarchica composta da elementi, geometrie e *layer* (livelli). I layer sono composti da geometrie, che a loro volta sono costituite da elementi.

1. Un elemento è il blocco di base di una geometria. I tipi di elementi spaziali supportati sono punti, linee e poligoni. Ad esempio, gli elementi possono modellare le costellazioni stellari (cluster di punti), strade (linee) e confini di contea (poligoni). Ogni coordinata in un elemento è memorizzata come una coppia X, Y. L'anello esterno e zero o più anelli interni (fori) di un poligono complesso sono considerati un singolo elemento. I punti sono costituiti da una coordinata, le linee sono costituite da due coordinate che rappresentano un segmento di linea dell'elemento e i poligoni sono formati da coppie di coordinate, una coppia di vertici per ogni segmento di linea del poligono. Le coordinate sono definite nell'ordine attorno al poligono (in senso antiorario per un anello poligonale esterno, in senso orario per un anello poligonale interno).

2. Una geometria è la rappresentazione di una caratteristica spaziale, modellata come un insieme ordinato di elementi primitivi. Una geometria può essere costituita da un singolo elemento, che è un'istanza di uno dei tipi primitivi supportati, o una raccolta di elementi omogenea o eterogenea. Un multi poligono, come quello usato per rappresentare un insieme di isole, è una collezione omogenea. Una raccolta eterogenea è quella in cui gli elementi sono di tipi diversi, ad esempio un punto e un poligono. Un esempio di geometria potrebbe descrivere il terreno edificabile in una città. Questo potrebbe essere rappresentato come un poligono con fori dove l'acqua o la lottizzazione ne impediscono la costruzione.

3. Un layer è una raccolta di geometrie con lo stesso set di attributi. Ad esempio, un livello in un GIS potrebbe includere caratteristiche

topografiche, un altro descrivere la densità di popolazione, mentre un terzo rappresentare la rete di strade e ponti nell'area (linee e punti). Le geometrie e l'indice spaziale associato per ogni livello sono memorizzati nel database in tabelle standard.

Nello specifico una geometria è una sequenza ordinata di vertici che sono collegati da segmenti di linea retta o archi circolari. La semantica della geometria è determinata dal suo tipo. Oracle Spatial and Graph supporta diversi tipi primitivi e composti di geometrie, tra cui anche tipi geometrici tridimensionali e quadridimensionali, in cui vengono utilizzate tre o quattro coordinate per specificare ciascun vertice dell'oggetto da definire. Come mostrato in figura 2.1, i tipi geometrici bidimensionali supportati sono:

- punti e cluster di punti;
- linee;
- poligoni;
- linee di archi (tutti gli archi sono generati come archi circolari);
- poligoni arco;
- poligoni composti;
- linee composte;
- cerchi;
- rettangoli ottimizzati.

I punti bidimensionali sono elementi composti da due ordinate, X e Y, che spesso corrispondono a longitudine e latitudine.
Le *linestring* (linee) sono composte da una o più coppie di punti che definiscono i segmenti di linea.
I poligoni sono composti da linestring collegate che formano un anello chiuso e l'area del poligono è implicita.

Ad esempio, un punto potrebbe rappresentare una posizione dell'edificio, una linestring una strada o una traiettoria di volo e un poligono uno stato, una città, un distretto di zona o un blocco di città.

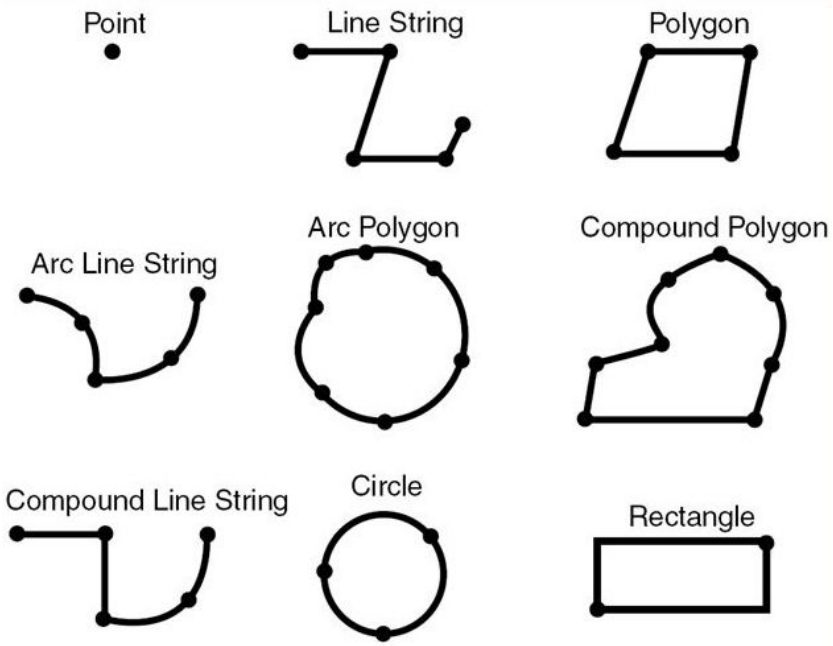


Figura 2.1: Tipi geometrici.

Fonte: https://docs.oracle.com/database/121/SPATL/geometry-types.htm#SPATL443

A supporto della manipolazione dei dati di cui sopra e dell'uso delle funzionalità di Oracle Spatial e Graph in Java, Oracle offre un'opportuna API [1].

2.3 Query spaziali

Oracle Spatial and Graph consente la formulazione di query spaziali di vario genere, attraverso è un modello di query a due livelli per risolvere query e join spaziali. Nello specifico vengono eseguite due operazioni distinte per risolvere

[1] https://docs.oracle.com/cd/E18283_01/appdev.112/e11829/toc.htm

una query, filtro primario e secondario, le quali combinate producono un set di risultati esatti.

1. Il filtro primario è considerato un filtro a basso costo, effettua una rapida selezione dei record per poi passare all'applicazione del secondario. Vengono confrontate le approssimazioni della geometria per ridurre la complessità di calcolo.

2. Il filtro secondario applica calcoli esatti alle geometrie risultanti dal primario, producendo una risposta precisa a una query spaziale. L'operazione da esso effettuata è dispendiosa dal punto di vista computazionale, ma viene applicata solo ai risultati del filtro primario e non all'intero set di dati.

L'operazione di filtro primario su un set di dati di input di grandi dimensioni, come mostrato in Figura 2.2, produce un set di candidati più piccolo, che contiene almeno il set di risultati esatti. L'operazione di filtro secondario sul set candidato più piccolo produce il set di risultati esatti.

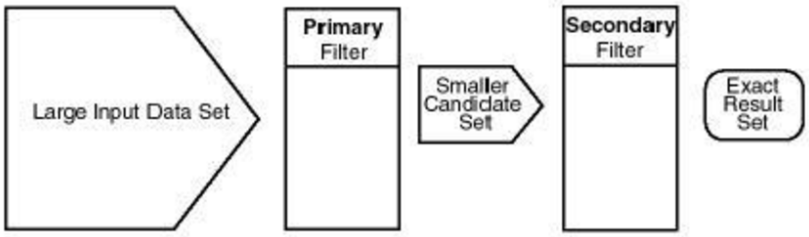


Figura 2.2: Modello di query spaziale, che mostra la relazione tra i filtri primario e secondario.

Fonte: https://docs.oracle.com/cd/B28359_01/appdev.111/b28400/query.gif

Oracle Spatial utilizza un indice spaziale per l'implementazione del filtro primario. Non è obbligatorio l'uso di entrambi i filtri, primario e secondario, in alcuni casi è sufficiente applicare il primario. Ad esempio, una funzione di zoom in un'applicazione di mappatura interroga i dati che presentano un'interazione con un rettangolo che rappresenta i limiti visibili. Il filtro principale

restituisce molto rapidamente un superset della query. L'applicazione può quindi applicare le routine di ritaglio per visualizzare l'area di destinazione.

Dunque lo scopo del filtro principale è quello di creare rapidamente un sottoinsieme di dati e ridurre il carico di elaborazione sul filtro secondario. Il filtro principale, quindi, dovrebbe essere per quanto possibile più efficiente, questo è determinato dalle caratteristiche dell'indice spaziale sui dati.

2.3.1 Indicizzazione di dati spaziali

L'indice spaziale è il fulcro di Oracle Spatial and Graph. Un indice spaziale, come qualsiasi altro indice, fornisce un meccanismo per limitare le ricerche, ma in questo caso esso si basa su criteri spaziali come l'intersezione e il contenimento. È necessario un indice spaziale per:

- trovare oggetti all'interno di uno spazio dati indicizzato che interagisce con un dato punto o area di interesse (window query);
- trovare coppie di oggetti all'interno di due spazi dati indicizzati che interagiscono spazialmente tra loro (spatial join).

2.3.2 Relazioni spaziali e filtri

Oracle Spatial and Graph usa il filtro secondario per determinare la relazione spaziale tra entità nel database, quest'ultima è basata su posizioni geometriche. Le relazioni spaziali più comuni si basano su topologia e distanza. Ad esempio, il confine di un'area è costituito da un insieme di curve che separa l'area dal resto dello spazio delle coordinate. L'interno di un'area consiste nei punti dell'area che non si trovano nel suo confine. Detto questo, due aree sono adiacenti se condividono parte di un confine ma non hanno in comune alcun punto nel loro interno.
La distanza tra due oggetti spaziali è la distanza minima tra qualsiasi punto in essi.

Per determinare le relazioni spaziali, Spatial ha diversi metodi di filtro secondari:

- l'operatore *SDO_RELATE* valuta i criteri topologici;

- l'operatore *SDO_WITHIN_DISTANCE* determina se due oggetti spaziali si trovano a una distanza specifica l'uno dall'altro;

- l'operatore *SDO_NN* identifica i *nearest neighbor* (punti più vicini) per un oggetto spaziale.

L'operatore SDO_RELATE implementa un modello a nove intersezioni per la categorizzazione delle relazioni topologiche binarie tra punti, linee e poligoni. Ogni oggetto spaziale ha un interno, un confine e un esterno. Il confine consiste di punti o linee che separano l'interno dall'esterno. Il confine di una linestring consiste nei sui punti finali, tuttavia se quest'ultimi si sovrappongono la linestring non ha confine. I confini di una linestring multipla sono i punti finali di ciascuna delle linestring che la compongono; nel caso in cui i punti finali si sovrappongono, vengono considerati solo quelli che si accavallano un numero dispari di volte. Il confine di un poligono è la linea che descrive il suo perimetro. L'interno è costituito da punti che si trovano nell'oggetto ma non sul suo confine, l'esterno è costituito da quei punti che non si trovano né nell'oggetto né suo confine.

Dato che un oggetto A ha tre componenti: un confine (Ab), un interno (Ai) e un esterno (Ae); ogni coppia di oggetti ha nove possibili interazioni tra i loro componenti. Le coppie di componenti hanno un'intersezione vuota (0) o non vuota (1). L'insieme di interazioni tra due geometrie è rappresentato, come nell'esempio in Figura 2.3, da una matrice a nove intersezioni che specifica quali coppie di componenti si intersecano e quali no.

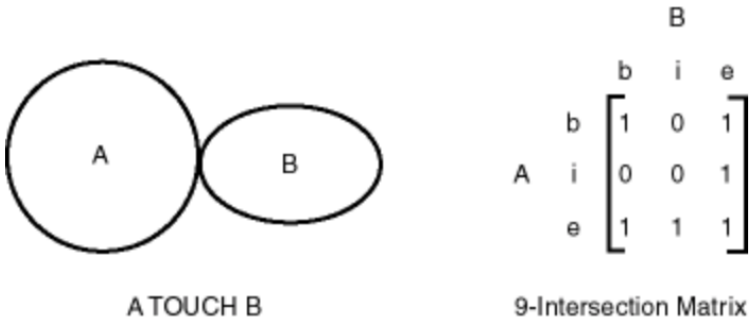


Figura 2.3: Modello a nove intersezioni, che mostra la maschera di bit associata a due oggetti (A e B) che hanno una relazione *TOUCH* (che verrà descritta in seguito).

Fonte: https://docs.oracle.com/cd/B28359_01/appdev.111/b28400/nine_inter.gif

Spatial utilizza i seguenti nomi per rappresentare i rapporti topologici, come rappresentato in Figura 2.4.

- *DISJOINT*: i confini e gli interni non si intersecano.
- *TOUCH*: i confini si intersecano ma gli interni no.
- *OVERLAPBDYDISJOINT*: l'interno di un oggetto interseca il confine e l'interno dell'altro, ma i due confini non si intersecano. Questa relazione si verifica, ad esempio, quando una linea ha origine fuori da un poligono e termina all'interno di esso.
- *OVERLAPBDYINTERSECT*: i confini e gli interni dei due oggetti si intersecano.
- *EQUAL*: i due oggetti hanno confini e interni uguali.
- *CONTAINS*: l'interno e il limite di un oggetto sono completamente contenuti all'interno dell'altro.
- *COVERS*: l'interno di un oggetto è completamente contenuto all'interno o al confine dell'altro oggetto e i loro confini si intersecano.

- *INSIDE*: il contrario di *CONTAINS*. A *INSIDE* B implica B *CONTAINS* A.
- *COVEREDBY*: l'opposto di *COVERS*. UN *COVEREDBY* B implica B *COVERS* A.
- *ON*: l'interno e il confine di un oggetto si trova sul confine dell'altro. Questa relazione si verifica, ad esempio, quando una linea si trova sul confine di un poligono.
- *ANYINTERACT*: gli oggetti non sono disgiunti.

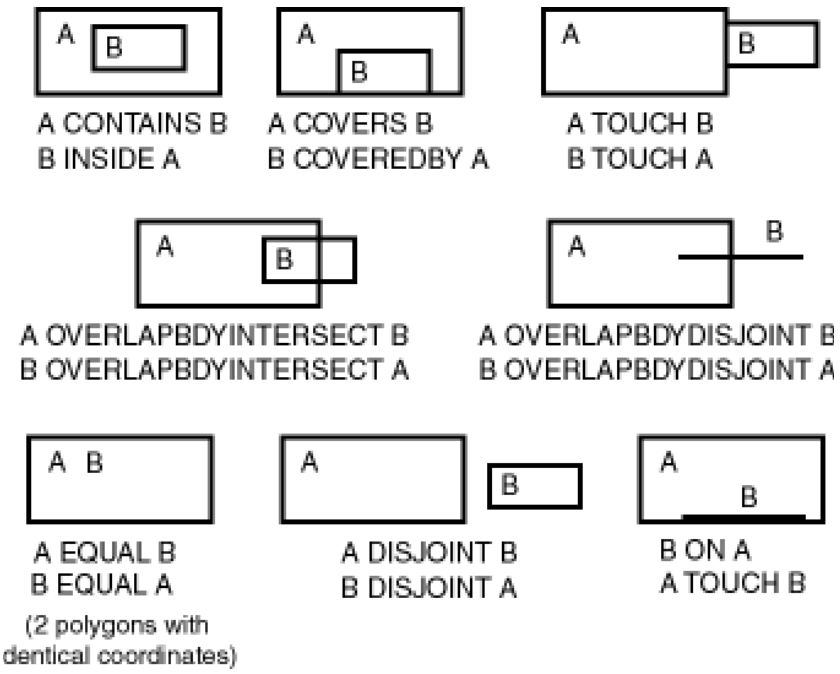


Figura 2.4: Relazioni topologiche precedentemente descritte.

Fonte: https://docs.oracle.com/cd/B28359_01/appdev.111/b28400/top_rel.gif

L'operatore SDO_WITHIN_DISTANCE determina se due oggetti spaziali, A e B, si trovano entro una distanza specifica l'uno dall'altro. Questo operatore costruisce dapprima un buffer di distanza, Db, attorno all'oggetto di riferimento B. Quindi controlla che A e Db non siano disgiunti. Il buffer di

distanza di un oggetto consiste di tutti i punti all'interno della distanza data da quell'oggetto. La Figura 2.5 mostra i buffer di distanza per un punto, una linea e un poligono.

Figura 2.5: Buffer di distanza per punti, linee e poligoni.

Fonte: https://docs.oracle.com/cd/B28359_01/appdev.111/b28400/buffers.gif

Nelle geometrie (punto, linea e poligono) mostrate nella Figura 2.5: le linee tratteggiate rappresentano i buffer di distanza. Notare come il buffer è arrotondato vicino agli angoli degli oggetti. La geometria sulla destra è un poligono con un foro: il rettangolo grande è l'anello poligonale esterno e il rettangolo piccolo è l'anello poligonale interno (il foro). La linea tratteggiata all'esterno del rettangolo grande è il buffer per l'anello esterno e la linea tratteggiata all'interno del rettangolo piccolo è il buffer per l'anello interno. L'operatore SDO_NN restituisce un numero specificato di oggetti da una colonna geometrica più vicina a una geometria specificata (ad esempio, i cinque ristoranti più vicini a un parco cittadino). Nel determinare quanto siano vicini due oggetti geometrici viene utilizzata la distanza più breve possibile tra due punti qualsiasi sulla superficie di ciascun oggetto.

Capitolo 3

La piattaforma

In questo capitolo viene presentata la piattaforma generale per l'elaborazione, arricchimento, visualizzazione e comprensione di dati di traiettoria.

3.1 Architettura

Il modello architetturale presentato punta a generalizzare il più possibile il concetto di estrazione di dati in qualunque ambito, esso è presentato in Figura 3.1. Sostanzialmente l'architettura si basa su due categorie di dati di partenza:

- *raw data* (dati grezzi): rappresentano un qualsiasi insieme di dati con caratteristiche spazio temporali che può riferirsi ad ambiti disparati, l'importante è che ci sia un timestamp con il relativo riferimento spaziale (geometria o latitudine e longitudine);

- *open data* (dati open): dati liberi che andranno ad arricchire i raw data.

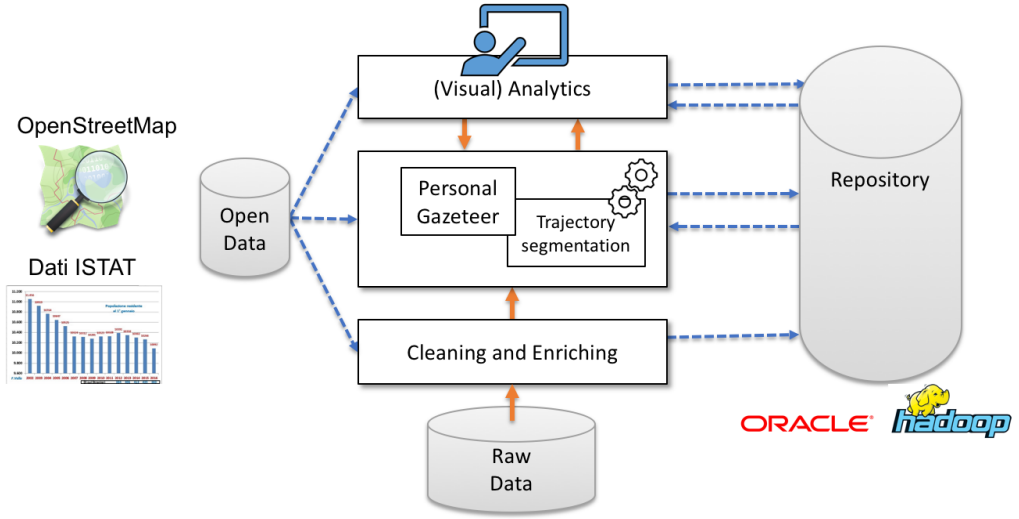


Figura 3.1: Schema del modello architetturale utilizzato, le linee blu rappresentano i flussi di informazioni da/verso fonti esterne, le linee arancioni il flusso di informazioni logiche.

La Figura 3.1 mostra che a partire dai raw data viene effettuato, interfacciandosi con gli open data, un processo di pulizia e arricchimento degli stessi il cui risultato viene trasferito nella *repository* (archivio). A questo punto vengono applicati dei *motorini* (engine) sui dati arricchiti, presenti nella repository, e gli open data, i quali si occupano di estrarre informazioni significative (che verranno trasferite nella repository) che possano aiutare a comprendere meglio i dati e a stabilire collegamenti semantici tra i dati stessi. Il livello superiore, attraverso scambi logici con quello inferiore, si occupa della visualizzazione e della analisi dei risultati ottenuti.

3.2 Il processo di arricchimento

In Figura 3.1 è presente un processo di pulizia e arricchimento dei dati. I dati grezzi, per loro natura possono contenere rumore, per cui prima di procedere al loro arricchimento si effettua una pulizia per ridurre al minimo le inconsistenze.

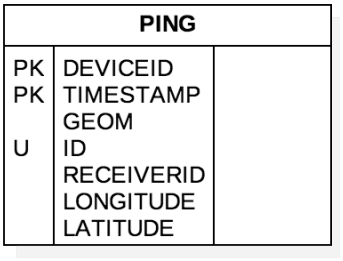


Figura 3.2: Modello relazionale generalizzato dei dati grezzi.

In primo luogo, dato che tra i dati raw potrebbero esserci più rilevazioni per lo stesso timestamp, bisogna eliminare i duplicati. Inoltre, poiché i dati potrebbero essere tanti, eccessivi per i nostri scopi, è opportuno filtrarli per intervalli temporali di un minuto. Il modello di partenza generalizzato dei dati grezzi è rappresentato in Figura 3.2.
Una volta effettuata la pulizia si può procedere all'arricchimento. In questa fase vengono aggiunte ai dati grezzi informazioni che saranno utili nelle successive elaborazioni, in sostanza queste informazioni possono essere suddivise in tre macrocategorie:

- **elaborazione dati**: una serie di dati derivati che velocizzeranno le successive elaborazioni. In generale, sono:
  - rappresentazione geometrica del punto o longitudine e latitudine, se non sono presenti nei dati di partenza;
  - un intero che rappresenta la rilevanza dell'oggetto nel dominio in termine di numero di punti raccolti per quell'oggetto, dunque l'oggetto con più punti avrà un identificativo di significatività pari a uno e quello con meno punti pari alla cardinalità di oggetti presenti;
  - tempo trascorso dalla rilevazione del punto precedente (cronologicamente) al corrente relativo ad uno stesso oggetto;

  - spazio percorso dal punto corrente al precedente (cronologicamente) di uno specifico oggetto;
  - velocità impiegata per muoversi dal punto precedente al corrente, relativamente ai punti ordinati di un singolo oggetto;
  - giorno della settimana della rilevazione;
  - ora di rilevazione.

- **visualizzazione dati**: informazioni necessarie ed utili per una corretta visualizzazione dei dati. Tra queste abbiamo:

  - rappresentazione della geometria del punto in formato GeoJSON, un formato standard di interscambio di dati geospaziali basato sul JSON;
  - un numero random che permetta di recuperare un certa percentuale di dati.

- **costruzione delle traiettorie**: a partire dalle informazioni aggiunte per l'elaborazione attraverso è possibile costruire le traiettorie, le quali verranno associate ai punti e avranno informazioni relative a:

  - tempo impiegato;
  - spazio totale percorso;
  - rappresentazione geometrica;
  - dati per la visualizzazione delle stesse (GeoJSON e numero random);
  - numero di punti contenuti;
  - tipologia di traiettoria (ove necessario).

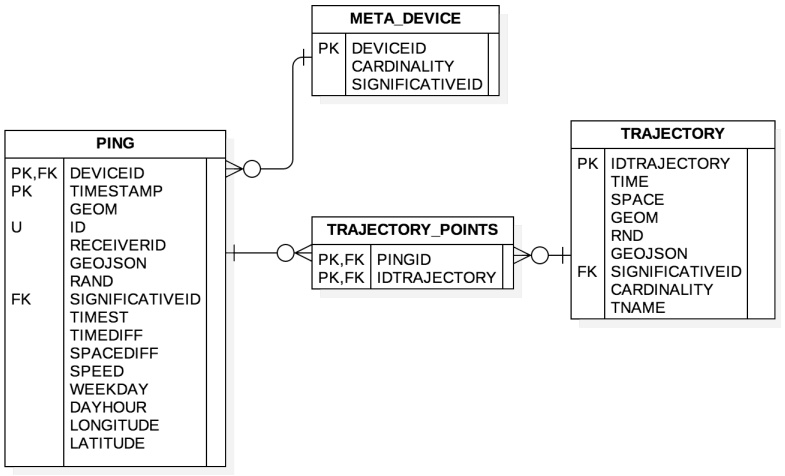


Figura 3.3: Modello relazionale dopo l'arricchimento dei dati.

Dopo il processo di arricchimento si arriva, dunque, allo schema relazionale in Figura 3.3.

3.3 I layer di dati

I layer, consistono essenzialmente nei vari livelli che vengono affiancati ai dati arricchiti. Fanno parte dei layer gli open data, per cui, a secondo dell'ambito di interesse, è possibile trovarne di vario genere. I layer aiutano ad immergere i dati arricchiti nel contesto in esame, valorizzandoli ulteriormente. Essi rendono possibile la costruzione d'intelligenze in grado di ricavare pattern frequenti e arricchire semanticamente i dati. In generale i layer sono:

- dati di *OpenStreetMap*: "una mappa, liberamente modificabile, del mondo intero, costruita praticamente dal nulla e rilasciata con una licenza libera"[1]. Sostanzialmente una comunità di mappatori contribuisce e mantiene i dati sulle strade, sentieri, caffè, stazioni ferroviarie e molto altro ancora, di tutto il mondo;

[1]https://wiki.openstreetmap.org - Visitato il 26.05.18

- *dati ISTAT*: dati provenienti dal Censimento della popolazione e delle abitazioni 2011, con ontologie di Basi Territoriali e di Dati Censuari[2];

- dati relativi ai quartieri o specifiche zone del territorio e dell'ambito d'interesse.

3.3.1 Engine per l'estrazione di informazione

I motorini mostrati in Figura 3.1 sono a supporto dell'interazione tra i layer e i dati arricchiti, essi estraggono informazioni significative relative alla correlazione tra questi dati. *Personal gazetteer*, basandosi sui concetti espressi nella Sezione 1.2, estrae gli stay point relativi all'oggetto del contesto in esame e costruisce un *"dizionario geografico personale"*. Attraverso delle interrogazioni spaziali tra gli stay point estratti e i layer è possibile collegare uno stay point a uno specifico dato di un layer. In questo modo anzichè dire "l'oggetto si trova frequentemente in un punto nello spazio" è possibile dire "l'oggetto si trova frequetemente in un punto nello spazio in prossimità di uno specifico punto di OpenStreetMap". Inoltre, è possibile costruire un'ontologia partendo da queste correlazioni tra dati arricchiti e open data, ovvero *una descrizione formale di un dominio del discorso* [1]. Questo concetto quindi offre un ulteriore passo avanti, ora è possibile dire, ad esempio, "l'oggetto si trova in una specifica categoria di luogo". Oltre a ciò per arricchire ulteriormente le informazioni ricavate è possibile generare statistiche sul tempo di permanenza in uno stay point (le quali verranno trattate nel Capitolo 4). Dopo tale processo il modello relazionale ottenuto è quello mostrato in Figura 3.4.

[2]http://datiopen.istat.it - Visitato il 26.05.18

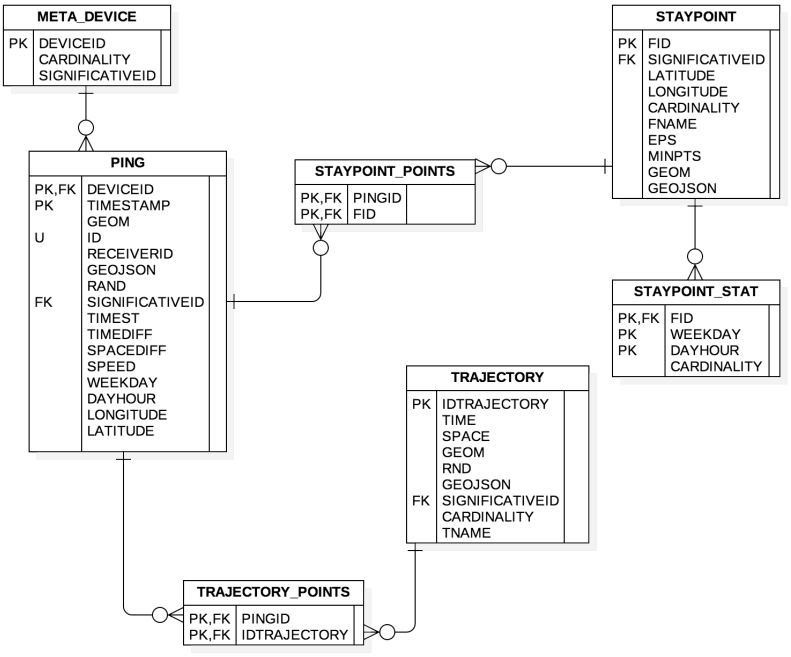


Figura 3.4: Modello relazionale generalizzato finale.

Capitolo 4

Tecniche per l'analisi dei dati

In questo capitolo ci addentriamo, dopo aver introdotto il modello di dati utilizzato, nelle tecniche impiegate per analizzare i dati e i pattern frequenti.

4.1 Analisi di dati navali

Negli ultimi decenni si è assistito a cambiamenti sostanziali in merito alla navigazione globale e al movimento delle merci per vie marittime, sia in termine di numero che di dimensioni delle navi. "Il trasporto marittimo è essenziale per l'economia mondiale poiché oltre il 90% del commercio mondiale è trasportato via mare ed è, di gran lunga, il modo più economico per spostare beni e materie prime in massa in tutto il mondo" (IMO - Organizzazione marittima internazionale)[1]. La tecnologia *AIS* (sistema di identificazione automatica) fornisce una grande quantità di informazioni quasi in tempo reale, si tratta di un sistema di messaggistica auto-segnalato originariamente concepito per evitare le collisioni per trasmettere dati relativi alla posizione delle navi [8]. Ci sono vari tipi di messaggi AIS che possono essere classificati in: informazioni statiche (nome, tipo, dimensione, ecc. della nave) e dinamiche (posizione in coordinate spaziali, velocità, rotta, direzione, destinazione, ora

[1] http://www.imo.org/en/OurWork/TechnicalCooperation/Documents/Brochure/English.pdf - Visitato il 15.05.2018

di arrivo stimata, ecc.) [11]. La frequenza con cui trasmettitori AIS inviano i dati varia a seconda della velocità dell'imbarcazione durante la navigazione, la trasmissione avviene ogni 2-10 secondi quando le navi sono in movimento e ogni 3 minuti mentre le navi sono ancorate [9]. In generale i dati includono:

- identificazione del servizio mobile marittimo della nave (MMSI): un identificativo univoco;
- stato di navigazione: "ancorata", "in corso utilizzando i motori" o "non sotto comando";
- tasso di svolta (destra o sinistra);
- SOG: velocità reale di avanzamento rispetto alla terra;
- accuratezza della posizione;
- longitudine e latitudine;
- COG: direzione della prua dell'imbarcazione rispetto alla terra;
- rotta effettiva;
- timestamp: in formato UTC (tempo coordinato universale) approssimato al secondo più vicino del momento in cui i dati sono stati generati.

Inoltre, i seguenti dati vengono trasmessi ogni 6 minuti (sia quando un'imbarcazione è in movimento che quando è ancorata):

- numero di identificazione della nave marittima dell'IMO: un numero che rimane invariato al momento del trasferimento della registrazione della nave in un altro paese;
- indicativo di chiamata internazionale assegnato alla nave dal suo paese di registrazione;
- nome della nave;

- tipo di nave/carico;
- dimensioni della nave;
- tipo di sistema di posizionamento, ad esempio GPS;
- posizione dell'antenna del sistema di posizionamento a bordo dell'imbarcazione;
- immersione della nave;
- destinazione;
- ora di arrivo stimata (ETA) alla destinazione.

4.2 Introduzione alla ricerca di pattern comportamentali

I dati AIS rappresentano una fonte fondamentale di informazioni, poiché attraverso la loro elaborazione è possibile ricavare varie conoscenze, tra cui [3]:

- informazioni sul "livello del traffico" in aree specifiche;
- estrarre le conoscenze per la previsione situazionale, ad esempio, scoprire automaticamente le zone di pesca sulla base di dati storici AIS trasmessi dai pescherecci [5];
- rilevare le anomalie, che possono essere causate da azioni illegali, come il contrabbando, l'inquinamento e la pesca non autorizzata nelle aree protette;
- classificare gli itinerari cercando di capire probabilità con cui una nave sta seguendo un certo itinerario [8];

- cercare di prevedere l'itinerario che una nave ha intenzione di percorre, in base alle conoscenze acquisite in precedenza e le informazioni della nave.

Per poter estrarre questo genere di conoscenze dalle traiettorie navali è necessario definire gli stay points delle imbarcazioni, attraverso uno specifico algoritmo di *clustering* (raggruppamento), e costruire il loro *"personal gazetteer"*, cercando di riadattare i concetti validi per lo studio dei comportamenti di una persona in questo specifico ambito.

4.3 Clustering

Il clustering è un processo di raggruppamento di oggetti tale che la similarità tra oggetti dello stesso gruppo (*cluster*) sia massimizzata e la similarità tra oggetti di gruppi diversi sia minimizzata [4]. Esistono vari approcci di clustering, uno di questi è basato sulla densità.

4.3.1 Clustering basato sulla densità

L'idea è quella di aggiungere l'area al cluster che è più vicino ad essa, considerando che la densità dei punti nell'area sia maggiore di una certa soglia [13]. I vantaggi di questo approccio rispetto ad altri tipi di clustering sono i seguenti:

- la forma dei cluster può essere arbitraria, un esempio è visibile in Figura 4.1;
- usando una soglia di densità i valori anomali, il rumore o i punti inusuali meno probabilmente verranno inseriti nella soluzione finale;
- il cambiamento dei parametri di ingresso richiesti è poco probabile;
- il risultato è deterministico: con uno stesso input l'output sarà sempre uguale.

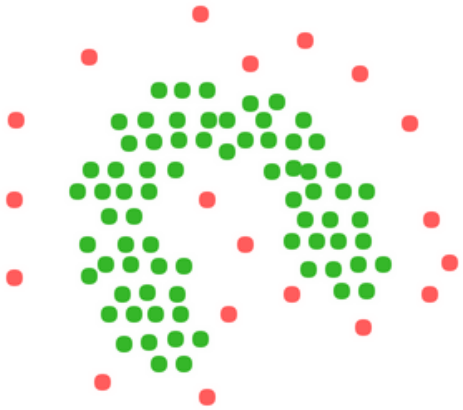


Figura 4.1: Un esempio di clustering basato sulla densità, dove i risultati hanno forma arbitraria.

DBSCAN

DBSCAN è un algoritmo di clustering basato sulla densità. Il suo funzionamento è il seguente; un punto p è un punto dati principale se almeno *MinPts* punti sono con distanza *Eps* da esso, e questi punti sono marcati come punti direttamente raggiungibili da p [13]. Il suo approccio è molto sensibile ai parametri di ingresso (*MinPts* e *Eps*), in alcuni casi l'algoritmo genererà un alto numero di punti all'interno della sua definizione di densità, ognuno dei quali potrebbe essere ulteriormente utilizzato per generare i propri punti raggiungibili dalla densità. E in questi casi, dunque, ci sarebbe un calo di performance e un elevato uso di memoria.

4.3.2 DjCluster

DjCluster è un altro algoritmo di clustering con un approccio basato sulla densità e sul *join* (unione). A differenza di DBSCAN che usa la nozione connessa dei grafi di cricca, DjCluster usa il concetto delle componenti connesse e risolve i problemi di performance di DBSCAN relativi all'alta sensibilità dei parametri di ingresso. Alla base di DjCluster c'è il concetto di *vicinato*

(neighborhood), il quale consiste nei punti con una certa distanza *Eps* e la condizione che questi punti siano almeno *MinPts*. Se non viene trovato nessun vicinato, il punto viene etichettato come rumore; altrimenti i punti formano un nuovo cluster se nessun vicino ha rapporti di vicinanza con altri cluster già esistenti, in caso contrario vengono inclusi nel cluster esistente [17].

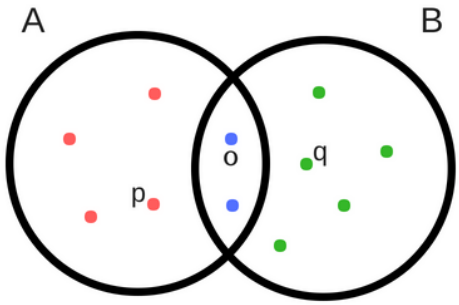


Figura 4.2: I cluster A e B sono di densità associabile, per la presenza del punto *o*.

Formalizzazione Il concetto di vicinato N basato sulla densità di un punto p, indicato come *N(p)*, è definito come:

$$N(p) = \{q \in S | dist(p, q) \leq Eps\}$$

dove S è il set di tutti i punti, q è un qualsiasi punto nel campione, *Eps* è il raggio del cerchio attorno a p che definisce la densità e *MinPts* è il minimo numero di punti richiesti all'interno del cerchio.
N(p) è definito come di densità associabile a *N(q)*, indicato come *J(N(p), N(q))*, rispetto a *Eps* e *MinPts*, se c'è un punto o che è contenuto sia in *N(p)* che *N(q)*. Un esempio di una reazione di densità associabile è mostrata in Figura 4.2.
Il cluster basato sulla densità e sull'associobilità C è definito come segue:

$$\forall p \in S, \forall q \in S, \exists N(p), N(q) \ tale \ che \ \exists J(N(p), N(q))$$

4.4 Il caso in esame

Nello specifico i dati AIS (o grezzi) presi in considerazione sono relativi ai primi quattro mesi del 2014 degli Stati Uniti[2]. Nel dettaglio il database è composto da 216 File Geodatabase (FGDB), ognuno dei quali rappresenta un mese di dati per una singola zona UTM degli Stati Uniti.

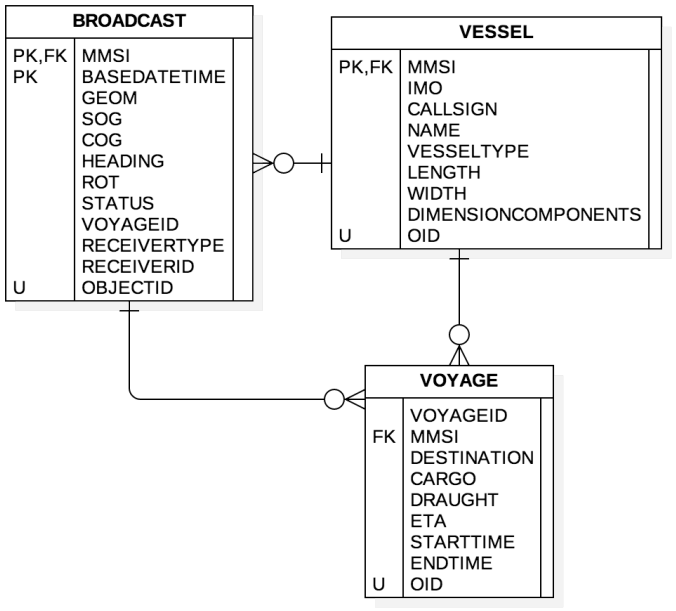


Figura 4.3: Modello relazionale dei dati di partenza.

Ogni file è formato da tre tabelle, seguendo teoricamente lo schema relazionale mostrato in Figura 4.3:

1. la tabella principale *Broadcast* (informazioni trasmesse), corrispondente ai dati grezzi presentati nel Capitolo 3, contiene le rilevazioni spazio-temporali delle imbarcazioni, che sono stati pre-filtrati per un intervallo temporale di un minuto;

2. la tabella *Vessel* (nave) contiene dati dettagliati delle navi;

[2]https://coast.noaa.gov/htdata/CMSP/AISDataHandler/2014/index.html - Visitato il 16.05.2018

3. la tabella *Voyage* (viaggio) contiene i report statistici dei vari viaggi.

A partire da tali dati si è effettuato il processo descritto nel Capitolo 3, quindi: l'arricchimento semantico, l'estrazione delle traiettorie e il clustering dei punti tramite l'algoritmo DjCluster.

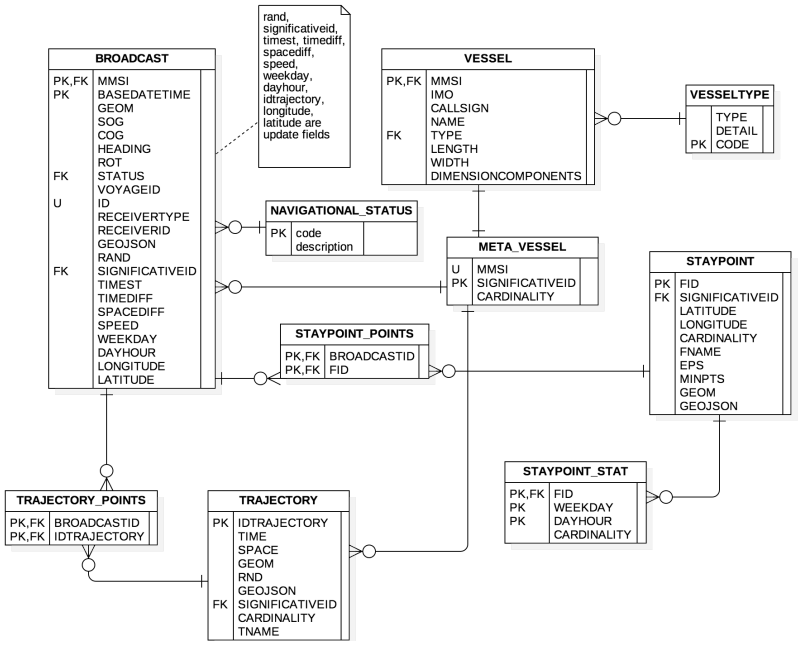


Figura 4.4: Modello relazionale del caso in esame.

Durante questi processi ci si è resi conto della presenza di inconsistenze e rumore nei dati, si è cercato di superare ciò mediante dei controlli e delle elaborazioni sui dati. Ad esempio abbiamo riscontrato la necessità di costruire traiettorie tramite concetti diversi da quelli utilizzati per segmentare le traiettorie nella tabella *Voyage* perchè la relazione tra le tabelle *Broadcast* e *Voyage* presenta in Figura 4.3 non risultava soddisfatta per molti dati, per cui non si riusciva a capire a quale viaggio appartenesse un determinato punto. Oltre a ciò nei si è rilevata presenza di rumore: ad esempio un'imbarcazione nel giro di pochi minuti si trovava in spazi a distanze superiori a quelle che realmente percorrerebbe. Perciò nella costruzione delle traiettorie si è fissato come parametro di divisione tra una traiettoria e un'altra non solo

il tempo ma anche lo spazio percorso. Dopo varie elaborazioni si è arrivati al modello relazionale in Figura 4.4. Nello schema in Figura 4.4 si può notare che la tabella *Broadcast* incorpora al suo interno tutti i campi corrispondenti all'arricchimento semantico, viene lasciato tutto in un'unica tabella per migliorare le performance di accesso ai dati. La dimensione della tabella è di circa *50 Gb*, poiché ci si è resi conto che l'accesso alla stessa, seppur attraverso un indice su chiave primaria, risultasse notevolmente lento si è proceduto con il partizionamento. La tabella è stata partizionata in *100 parti* usando come criterio di partizionamento percentili di MMSI, in questo modo si sono ottenuti cento range di imbarcazioni equamente distribuiti. Attraverso tale tecnica si sono ridotti notevolmente i tempi di risposta da parte del database Oracle.

Seguendo l'architettura presentata nel Capitolo 3, per questo specifico ambito i layer aggiuntivi consistono in:

- **porti** [3]: ubicazione e caratteristiche fisiche, le strutture e servizi offerti dai principali porti e terminal di tutto il mondo;

- **confini** [4]: limiti e i confini marittimi degli Stati Uniti;

- **zone navigabili, di commercio e corsie** [5]: zone di spedizione che delineano attività e regolamenti per il traffico marittimo delle navi, corsie di traffico che definiscono il flusso del traffico e aree in cui le navi devono navigare con cautela, rotte consigliate, aree da evitare e soggette a limiti.

[3]https://msi.nga.mil/NGAPortal/MSI.portal?_nfpb=true&_pageLabel=msi_portal_page_62&pubCode=0015 - Visitato il 28.05.2018

[4]https://nauticalcharts.noaa.gov/data/us-maritime-limits-and-boundaries.html#access-digital-data - Visitato il 28.05.2018

[5]https://catalog.data.gov/dataset/shipping-fairways-lanes-and-zones-for-us-waters44831 - Visitato il 28.05.2018

4.4.1 Ricerca di pattern

Sulla base di quanto descritto nella Sezione precedente, si ricercano i seguenti pattern comportamentali:

1. luoghi frequentati dalle navi;
2. analisi delle traiettorie percorse;
3. possibili nuovi pattern: statistiche relative agli stay point.

I quali sono da confrontare con la *verità di fondo* (ground truth) rapportata alle categorie di navi.

Ricerca e analisi dei luoghi frequenti

Definizione è considerato *luogo frequente* quel luogo, appartenente ad un determinato layer aggiuntivo, con il quale un'imbarcazione ha almeno uno stay point che ha una determinata interazione con esso.
Le interazioni possono essere sostanzialmente della stessa tipologia delle relazioni spaziali descritte nella Sezione 2.3.2. Possibili combinazioni di queste interrogazioni nell'ambito marittimo potrebbero essere:

- **ricerca dei porti frequenti**, attraverso un'interrogazione del tipo nearest neighbor (operatore *SDO_NN* in Oracle) tra le geometrie dei porti e quelle degli stay point (ricavati attraverso l'applicazione di un algoritmo di clustering). In questo caso è necessario definire una soglia di vicinanza, nel contesto specifico un parametro accettabile è *2 km*. Confrontando i risultati per categoria di imbarcazione e controllando i tempi in cui una nave è in un porto è possibile ricavare dei comportamenti frequenti. Un modello è ottenuto confrontando il numero di porti frequenti ottenuti per imbarcazione e paragonandoli con la tipologia della stessa è possibile capire se i dati effettivamente rispecchiano la verità di fondo. Un esempio pratico: "presumibilmente un'imbarcazione che frequenta tanti porti sarà un rimorchio, ma difficilmente sarà

un peschereccio il quale si presume che sia in pochi porti". Un altro modello è quello di capire il tempo effettivo che un oggetto trascorre in un porto o in generale nei porti, in modo da poter capire di conseguenza anche quanto tempo trascorre in mare.

- **ricerca delle zone frequentate**, in questo caso si procede con un'interrogazione per ricercare l'interazione tra un punto e un poligono (operatore *SDO_RELATE* in Oracle) perché bisogna cercare tutti i poligoni con cui gli stay point hanno avuto un'interazione. Attraverso questi risultati è possibile fare un ragionamento simile a quello fatto nei porti. Ad esempio, una nave che avrà interazioni con molte zone potrebbe essere un'imbarcazione ausiliaria o nave pilota.

Analisi delle traiettorie

La creazione delle traiettorie avviene attraverso l'algoritmo sviluppato [6]. L'**input** dello stesso consiste in un set di punti arricchiti di una specifica imbarcazione ordinati cronologicamente. Ogni punto contiene delle informazioni riguardanti il rapporto tra esso e il precedente, come lo *spazio* (distance), il *tempo* (timeDiff) e la *velocità* (speed) che intercorrono tra essi.
Con *tname*, viene identificato il tipo di traiettoria, si parla di traiettorie *stazionarie* e *in movimento* (stationary e move). Una traiettoria è considerata stazionaria se la velocità in una coppia di punti è inferiore a un certo parametro (es. 2 km/h), altrimenti è in movimento.
L'algoritmo aggiungere punti incrementalmente alla traiettoria corrente se e solo se vengo rispettate le condizioni che dipendono da tre parametri: *speedThreshold, timeThreshold, distanceThreshold*, i quali indicano rispettivamente le soglie di velocità, tempo e spazio considerati sufficienti al fine di proseguire o meno una traiettoria. Nel caso in cui non rispettata una delle condizioni la traiettoria costruita viene inserita nel database e si genera una nuova traiettoria che sarà formata dal punto corrente. In Figura 4.5 sono mostrati tre esempi di traiettoria che seguono l'andamento

[6]

spazio/temporale. L'algoritmo crea la prima traiettoria, formata dai punti $\{p1, p2, p3, p4, p5, p6\}$, dato che $distance(p6, p1') > distanceThreshold$ crea la seconda traiettoria formata dai punti $\{p1', p2', p3', p4', p5'\}$. A questo punto dato che $timediff(p1', p1'') > timeThreshold$ l'algoritmo crea la terza traiettoria formata per il momento da $\{p1''\}$.

Algorithm 1 Estrazione di traiettorie

Input:

- *points* - set ordinato cronologicamente di punti arricchiti di un'imbarcazione;
- *speedThreshold* - soglia di velocità;
- *timeThreshold* - soglia di tempo;
- *distanceThreshold* - soglia di spazio;

$trajectory \Leftarrow Trajectory(points.first.from)$
for $point \in points$ **do**
  if $point.speed < speedThreshold$ **then**
    $tname \Leftarrow STATIONARY$
  else
    $tname \Leftarrow MOVE$
  end if
  if $tname = trajectory.tname$
  and $point.timeDiff \leq timeThreshold$
  and $point.distance \leq distanceThreshold$ **then**
    $trajectory.put(point.to)$
  else
    $insert(trajectory)$
    $trajectory \Leftarrow Trajectory(point.to)$
  end if
end for

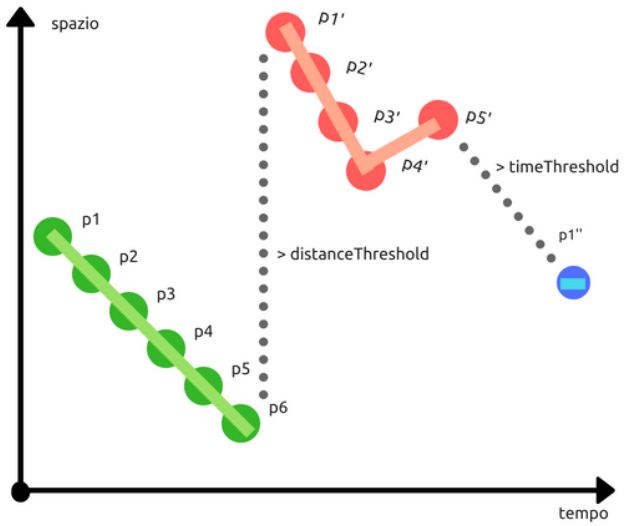


Figura 4.5: Esempio di traiettorie costruite in base ai parametri di soglia spazio temporale.

Ogni volta che si aggiunge un punto ad una traiettoria si aggiornano dei dati derivati che riguardano lo spazio, il tempo e la cardinalità della traiettoria stessa. Attraverso questi dati è possibile costruire statistiche sia per imbarcazione che per tipologia e confrontare il rapporto tra queste categorie di dati. Il confronto di questi dati, può avvenire ad esempio attraverso le formulazioni matematiche di media e deviazione standard e il relativo riscontro grafico delle stesse. Dunque, ci si aspetta che ci siano categorie di imbarcazioni che effettueranno tratte più lunghe e a velocità più sostenute rispetto ad altre.

Possibili nuovi pattern: statistiche relative agli stay point

Dai risultati ottenuti a seguito dell'applicazione dell'algoritmo di clustering è possibile costruire delle statistiche relative alle fasce orarie, i giorni della settimana e la cardinalità con un cui un'imbarcazione frequenta un determinato cluster. Un esempio di tali statistiche è mostrato nella Tabella 4.1,

dalla quale è possibile capire che in determinati giorni ci sono fasce orarie in cui l'imbarcazione frequenta uno o più luoghi e in altre, invece, non starà per tempo sostanziale in un luogo, queste vengono delineate con una cella vuota.

Tabella 4.1: Esempio di tabella con le statistiche temporali relative ai cluster di una determinata imbarcazione

#	**Lunedì**	**Martedì**	**Mercoledì**	**Giovedì**	**Venerdì**	**Sabato**	**Domenica**
0:00 - 7:00	Cluster 1 cardinalità 1000	Cluster 1 cardinalità 500	Cluster 2 cardinalità 1000	Cluster 1 cardinalità 1000	Cluster 2 cardinalità 1000	Cluster 3 cardinalità 3000	Cluster 1 cardinalità 1000
8:00 - 12:00	Cluster 2 cardinalità 200	Cluster 2 cardinalità 1000	Cluster 1 cardinalità 1500	Cluster 1 cardinalità 700 Cluster 2 cardinalità 2000	Cluster 1 cardinalità 1000	Cluster 1 cardinalità 1000	Cluster 3 cardinalità 1000
13:00 - 18:00	Cluster 3 cardinalità 400	Cluster 1 cardinalità 700	Cluster 1 cardinalità 100 Cluster 4 cardinalità 300				
19:00 - 23:00					Cluster 1 cardinalità 1000		Cluster 3 cardinalità 1000

Questo tipo di statistiche si potrebbero associare alle navi o alle zone in modo da capire in che giorni e in quali fasce orarie una certa imbarcazione frequenta determinati luoghi e con quale frequenza. Ad esempio, ci si può aspettare che una nave di una certa categoria frequenti i posti in determinati periodi e difficilmente in altri.

Capitolo 5

Il prototipo realizzato

In questo capitolo viene presentato il prototipo realizzato nel progetto di tesi.

5.1 Interfaccia

L'interfaccia grafica consiste in un modello basato su più layer, attraverso ognuno dei quali è possibile percepire nella mappa determinate caratteristiche dei dati utilizzati.

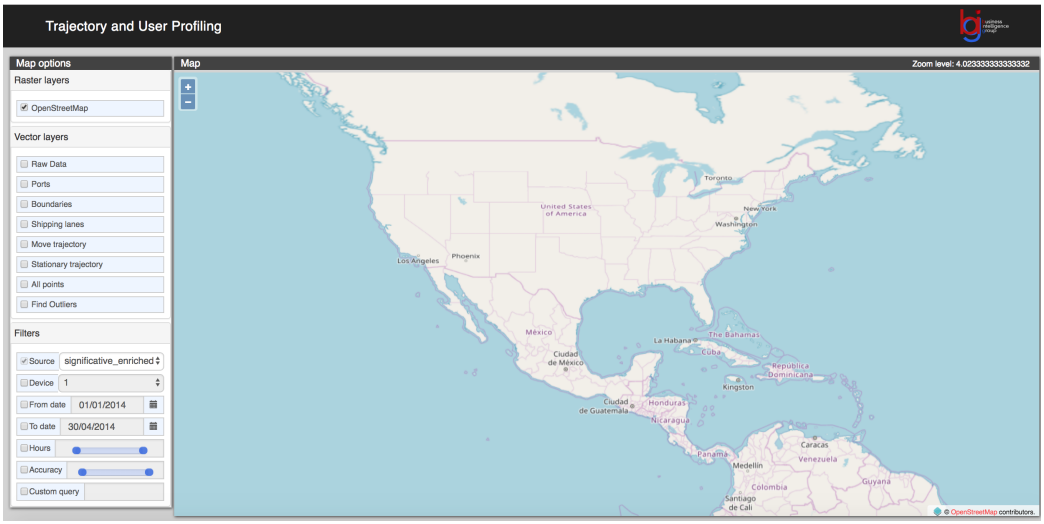


Figura 5.1: Interfaccia del prototipo realizzato.

La Figura 5.1 rappresenta l'interfaccia relativa al prototipo realizzato, com'è possibile notare a sinistra della Figura, ci sono due categorie di layer: *layer raster* e *layer vettoriali*. Questi ultimi forniscono un modo per rappresentare le *caratteristiche* (feature) del mondo reale all'interno dell'ambiente GIS, attraverso attributi che consistono in testo o informazioni numeriche che descrivono tali caratteristiche. Una caratteristica ha una sua rappresentazione geometrica (punto, linestring o poligono). I layer raster sono, invece, composti da matrici di pixel, dove ogni cella contiene un valore che rappresenta le condizioni dell'area da essa coperta. I dati raster vengono usati nelle applicazioni GIS quando si vogliono visualizzare delle informazioni che sono continue lungo un'area e che non sono facilmente divisibili in oggetti vettoriali[1]. Per queste ragioni il layer raster dell'applicazione è la base della mappa in OpenStreetMap e, al contrario, tutti i layer che andranno ad arricchire la mappa sono layer vettoriali. Nel dettaglio questi ultimi rappresentano:

- *raw data*: dati grezzi di partenza;
- *ports* (porti): dati relativi ai porti vicini ai dati grezzi;
- *boundaries* (confini): confini degli Stati Uniti;
- *shipping lanes* (corsie di navigazione): zone navigabili, di commercio e corsie presenti negli Stati Uniti;
- *move trajectory* (traiettorie in movimento): traiettorie in cui le imbarcazioni sono considerate in movimento (velocità $>$ 2km/h);
- *stationary trajectory* (traiettorie ferme): traiettorie in cui le imbarcazioni sono considerate ferme (velocità $<$ 2km/h);
- *all points* (stay point): stay point delle imbarcazioni;
- *find outliers* (trova valori anomali): ricerca di valori anomali (superiormente o inferiormente) rispetto alle lunghezze medie delle traiettorie della categoria.

[1]https://docs.qgis.org/2.8/it/docs/gentle_gis_introduction/ - Visitato il 30.05.2018

Ove previsto, è possibile filtrare le informazioni di uno specifico layer in base a: una specifica imbarcazione, un intervallo di date, una fascia oraria o in modo personalizzato.

5.2 L'analisi dei dati

In questa sezione vengono presentati i risultati ottenuti in merito alla ricerca dei pattern presentati nel precedente capitolo.

5.2.1 Ricerca e analisi dei luoghi frequenti

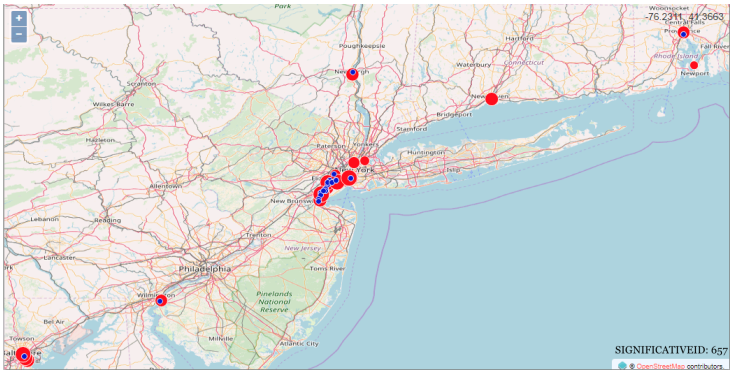
Per quanto riguarda la ricerca dei luoghi frequentati si sono svolte due ricerche differenti: una in merito ai porti e l'altra alle zone navigabili, di commercio e corsie.

Ricerca e analisi dei porti frequenti

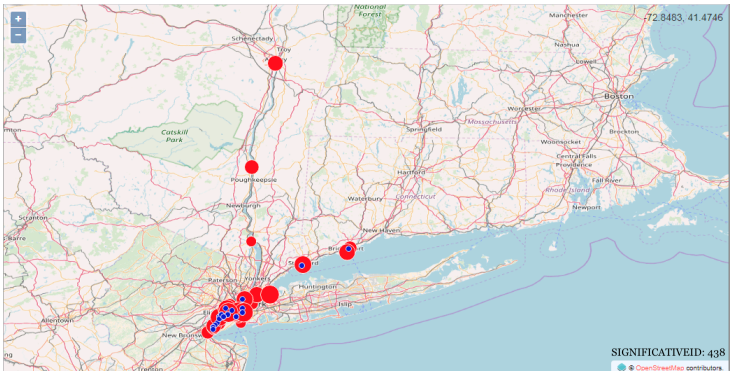
In Figura 5.2 sono rappresentati gli stay point e i porti frequenti delle prime tre barche con maggior numero di porti frequentati. Dalla figura è possibile notare che queste tre navi si trovano tutte nella zona di New York, dove la denisità dei porti è maggiore. Le prime due navi (5.2a e 5.2b) sono dei rimorchiatori, per la terza (5.2c), invece, non è specificata la categoria.
Dai risultati, si è notato che ci sono delle navi che non frequentano alcun porto poiché per queste non esistono stay point a una distanza al massimo di 2km da un porto.

Ricerca e analisi delle zone frequenti

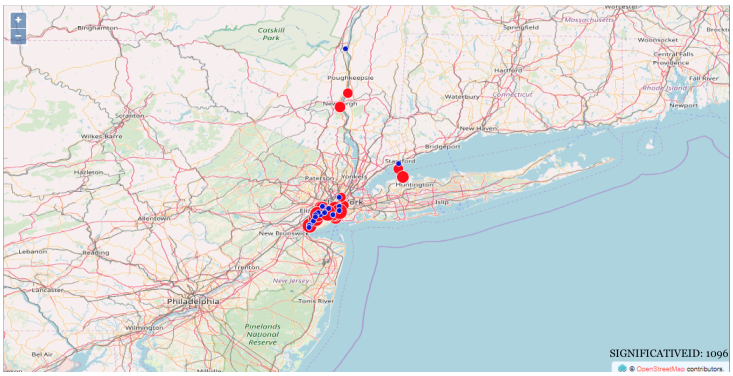
Dai risultati emerge che la prima imbarcazione (5.3a) è una nave pilota che nella zona di San Francisco, dalla quale esce e rientra varcando il confine e senza fermarsi nelle acque al di fuori del confine per periodi consistenti. Infatti svolge quello che dovrebbe fare una nave di questa categoria, ovvero pilotare altre navi in porto o trasportare il pilota su quelle da guidare sino al



(a) MMSI: 366294450 - 22*123m - rimorchiatore.

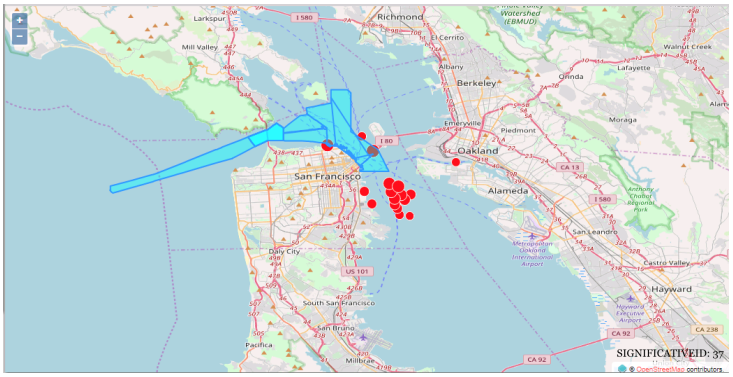


(b) MMSI: 367005056 - 10*31m - rimorchiatore.

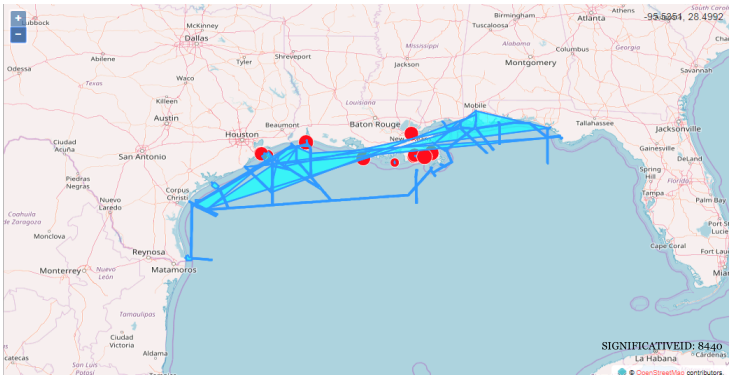


(c) MMSI: 367506466 - 10*31m - categoria non specificata.

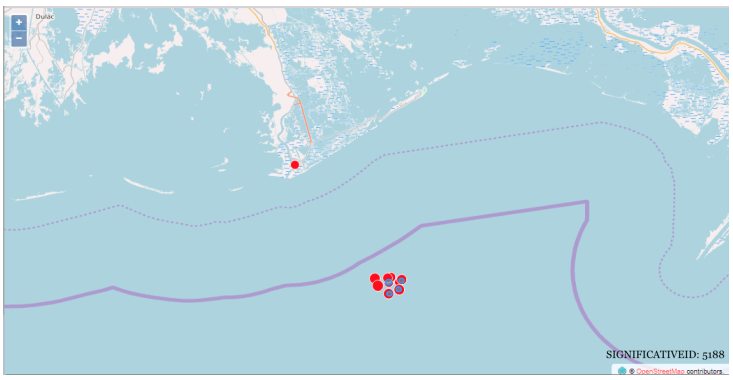
Figura 5.2: Le tre barche che frequentano il maggior numero di porti. In blu sono evidenziati i porti e in rosso gli stay point, i quali hanno un raggio del cluster dimensionato in base alla cardinalità dello stesso.



(a) MMSI: 367000173 - 5*23m - nave pilota.



(b) MMSI: 338030059 - dimensioni non specificate - imbarcazione ausiliaria.



(c) MMSI: 367202207 - 9*55m - categoria riservata.

Figura 5.3: Le tre imbarcazioni che hanno interazioni con il maggior numero di aree e corsie marittime. In celeste sono evidenziate le aree e in rosso gli stay point, i quali hanno un raggio del cluster dimensionato in base alla cardinalità dello stesso.

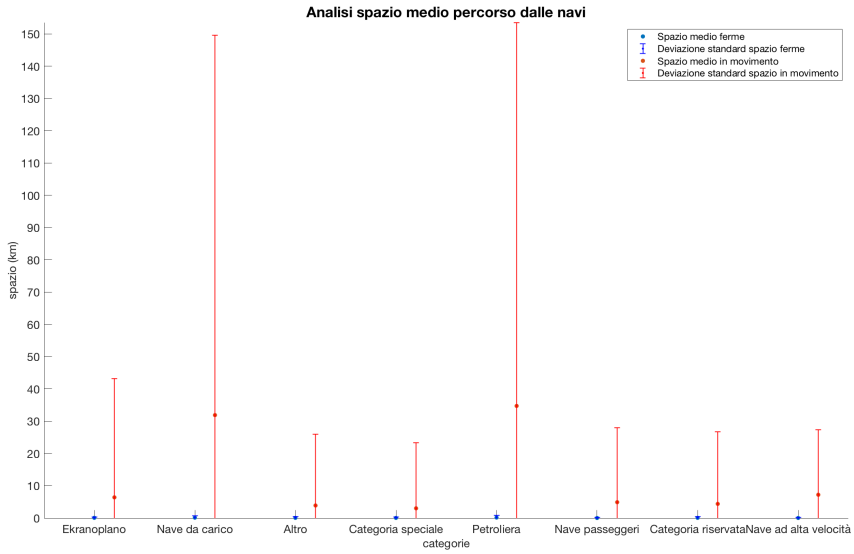
suo ormeggio in porto. La seconda (5.3b) è un'imbarcazione ausiliaria infatti svolge il proprio all'interno delle zone costiere. La terza imbarcazione (5.3c) appartiene ad una categoria riservata, è interessante come si sposti sempre nelle stesse quattro zone tre delle quali recitano 'Evitare l'ancoraggio in prossimità delle condotte marine, l'ancoraggio vicino a queste linee sommerse può causare danni all'ancora o alle condutture'.

5.2.2 Analisi delle traiettorie

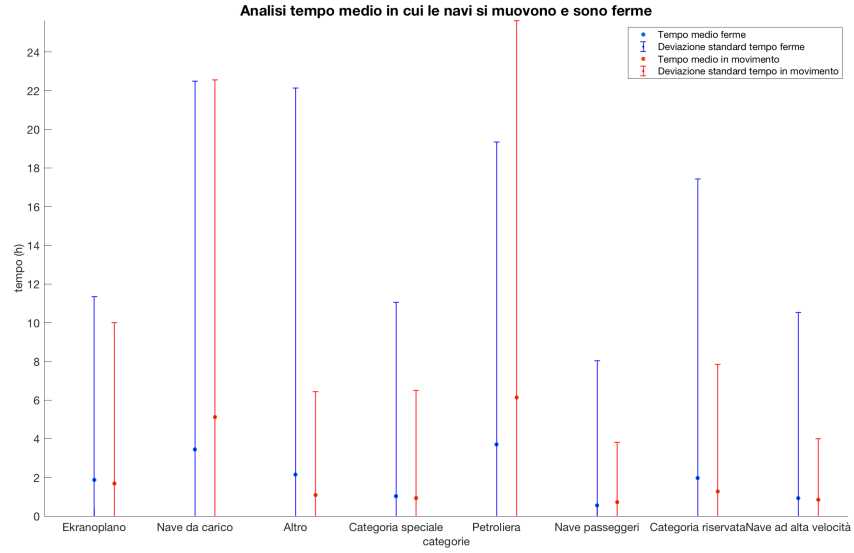
In merito all'analisi delle traiettorie si è proceduto con la costruzione di grafici relativi alla media e alla deviazione standard dello spazio e del tempo delle traiettorie raggruppando i dati in base alle categorie di appartenenza delle imbarcazioni in base allo standard AIS. Le quali nello specifico sono:

- *ekranoplano*: un mix tra idrovolante e aliscafo;
- *nave da carico*: navi che trasportano merci;
- *altro*: categorie ulteriori;
- *categoria speciale*;
- *petroliera*: imbarcazione adibita al trasporto di petrolio e derivati;
- *nave passeggeri*: nave adibita al trasporto di persone;
- *categoria riservata*;
- *nave ad alta velocità*: nave ad alta velocità per uso civile.

Un parametro importante da notare è che la deviazione standard tra queste medie risulta consistente, considerando le varie sotto-categorie di questi gruppi hanno differenze notevoli. I risultati ottenuti in merito alle analisi spaziali sono mostrati nel grafico in Figura 5.4a, come mostrato nella legenda, in blu sono rappresentati media e deviazione standard per lo spazio considerato come traiettoria stazionaria e in rosso media e deviazione standard per quello



(a)



(b)

Figura 5.4: Analisi dello spazio (5.4a) del tempo (5.4b) medio delle traiettorie raggruppati per categoria di imbarcazione..

in movimento. Da quanto esposto nel grafico è possibile percepire che le imbarcazioni che si muovono per più km sono le navi da carico e le petroliere che hanno traiettorie di una media di 35 km. Le altre categorie percorrono tutte circa 10 km.

Per quanto riguarda l'analisi temporale invece la Figura 5.4b mostra ciò che accade, in blu vengono indicate media e deviazione standard per il tempo delle traiettorie considerate ferme e in rosso media e deviazione standard per quelle in movimento. Da questo grafico emerge che le categorie che trascorrono maggior tempo in movimento sono le navi da carico e le petroliere, come ci si aspettava dai risultati ottenuti nel precedente grafico, le quali trascorrono in media 5/6 ore in movimento. Inoltre le altre categorie hanno tempi ferme e in movimento uguali, eccetto quelle riservate e le "altre" per cui il tempo in cui sono ferme supera quello in movimento. Questi risultati, come è già stato detto, sono molto generali e raggruppano al loro interno molte sotto-categorie, per capire più dettagliatamente e più precisamente cosa accade bisognerebbe svolgere lo stesso lavoro su tali sotto-gruppi.

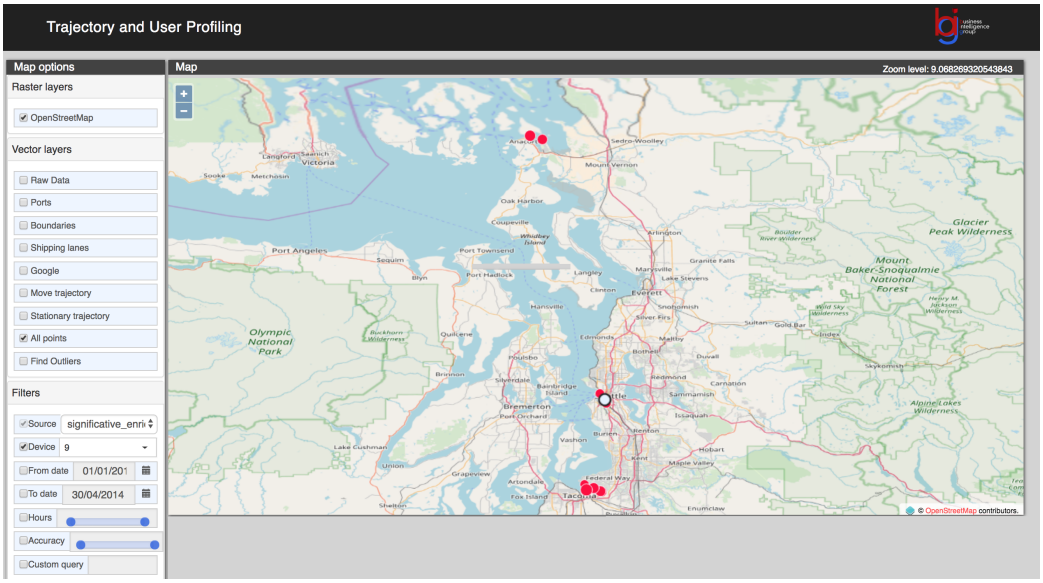
Figura 5.5: Anomalie relative a una porzione dei dati per la categoria nave passeggeri.

Partendo dai grafici sopra esposti, il layer *find outliers* ricerca le anomalie, classificando le traiettorie come senza anomalia, con anomalia superiore/inferiore in base alla lunghezza media delle traiettorie e il rapporto della stessa con la lunghezza media delle traiettorie della categoria di appartenenza. In

Figura 5.5 è mostrato un esempio di anomalie relative ad una porzione di dati per la categoria di navi passeggeri. In verde sono evidenziate le traiettorie con valori sotto al 30% della lunghezza media delle traiettorie di tale categoria, in blu quelle che superano il 30% e in rosso quelle che rientrano nello standard.

5.2.3 Statistiche relative agli stay point

In merito alle statistiche relative agli stay point, si è costruita una tabella all'interno del prototipo. Come mostrato in Figura 5.6, quando viene selezionato il layer degli stay point con il filtro su un'imbarcazione è possibile visualizzare la tabella con le statistiche relative ai giorni della settimana, fasce orarie e cardinalità di frequenza degli stay point dell'imbarcazione selezionata. Dunque nella cella corrispondente ad una specifica fascia oraria ed uno specifico giorno è possibile visualizzare quali cluster frequenta l'imbarcazione selezionata in quel periodo di tempo e con quale cardinalità. Ad esempio: *il lunedì dalle 8:00 alle 12:00 l'imbarcazione selezionata frequenta i cluster 1 e 2, rispettivamente con cardinalià 724 e 265.*
Inoltre quando l'utente seleziona uno specifico cluster lo stesso viene evidenziato nella mappa e nella tabella vengono indicate le altre celle in cui esso compare. In questo modo è possibile capire istantaneamente i luoghi frequentati maggiormente da un'imbarcazione e la loro collocazione temporale.



#	Lunedì	Martedì	Mercoledì	Giovedì	Venerdì	Sabato	Domenica
0:00 - 7:00	cluster 1 *card=1267* cluster 2 *card=367* cluster 4 *card=87* cluster 7 *card=65*	cluster 1 *card=1144*	cluster 1 *card=1423* cluster 2 *card=338* cluster 5 *card=56*	cluster 1 *card=1597* cluster 2 *card=748* cluster 8 *card=53* cluster 6 *card=37*	cluster 1 *card=1636* cluster 2 *card=84*	cluster 1 *card=1793* cluster 2 *card=635*	cluster 1 *card=1464* cluster 2 *card=323*
8:00 - 12:00	cluster 1 *card=724* cluster 2 *card=265*	cluster 1 *card=872*	cluster 1 *card=1183* cluster 2 *card=171*	cluster 1 *card=992* cluster 2 *card=354*	cluster 1 *card=1293* cluster 12 *card=15*	cluster 1 *card=1141* cluster 2 *card=219*	cluster 1 *card=691* cluster 2 *card=522*
13:00 - 18:00	cluster 1 *card=784* cluster 2 *card=255*	cluster 1 *card=1196* cluster 2 *card=83*	cluster 1 *card=1191* cluster 2 *card=104*	cluster 1 *card=1341* cluster 2 *card=215*	cluster 1 *card=955*	cluster 1 *card=840* cluster 2 *card=242*	cluster 1 *card=778* cluster 2 *card=535*
19:00 - 23:00	cluster 1 *card=1027* cluster 10 *card=42* cluster 2 *card=15*	cluster 1 *card=695* cluster 2 *card=126* cluster 3 *card=112*	cluster 1 *card=1046*	cluster 1 *card=606* cluster 2 *card=89*	cluster 1 *card=817* cluster 3 *card=99* cluster 2 *card=96* cluster 11 *card=37*	cluster 1 *card=889*	cluster 1 *card=846* cluster 2 *card=258*

Figura 5.6: Riscontro grafico delle statistiche sugli stay point

Conclusioni e sviluppi futuri

Il lavoro di tesi esposto ha avuto come obiettivo principale quello di estrarre informazioni significative dalle tracce di movimento grezze raccolte, in questo caso specifico, dalle imbarcazioni che hanno navigato nei primi tre mesi del 2014 nelle acque degli Stati Uniti.
Il primo step è stato quello di effettuare una pulizia e un arricchimento dei dati, sui quali successivamente si sono applicati gli algoritimi di ricerca degli *stay point* e di *costruzione delle traiettorie*. I risultati di tutti i livelli che si sono venuti a creare in queste fasi sono visualizzabili agevolmente nella piattaforma web sviluppata.

I risultati conseguiti sono molteplici: in primo luogo si è capito che l'architettura e gli algoritmi utilizzati possono essere riadattati ad una molteplicità di ambiti, attraverso dei piccoli accorgimenti dovuti al dominio applicativo. Ad esempio, nel nostro caso di studio un'imbarcazione è considerata vicino ad un porto se è ad un raggio di 2km dallo stesso, per una persona sicuramente questo parametro va rivisto ed abbassato a 50m. Inoltre dai risultati grafici ottenuti nella visualizzazione delle statistiche degli *stay point*, mostrati in Figura 5.6, è possibile (in ogni ambito) capire a colpo d'occhio quali sono i punti frequentati maggiormente da un oggetto, in quale fascia oraria e in quale giorno della settimana. In conclusione si può affermare che il progetto sia stato portato a termine con successo, raggiungendo gli obiettivi prefissati.

Considerando quanto sopra esposto, primo tra gli sviluppi futuri di questo progetto è quello di far in modo che la piattaforma sia facilmente modificabile sia fronte di cambiamenti di ambito, sia di aggiunta di specifiche. Inoltre,

come mostrato in Figura 3.1 e considerando le grosse di quantità di dati che potrebbero essere gestiste, risulta facile effettuare una migrazione verso una piattaforma totalmente big data, come *Hadoop*. Dal punto di vista algoritmico, invece, quanto sviluppato può essere integrato a successivi step, come il clustering delle traiettorie che, integrato al concetto di *stay point*, incrementerebbe l'accuratezza della ricerca dei pattern. Oltre a questo, uno sviluppo potrebbe essere quello di evidenziare nella piattaforma web, oltre che i singoli punti, anche informazioni sulle imbarcazioni e sulle loro direzioni di movimento; in generale degli oggetti della piattaforma e delle loro direzioni di movimento.

Bibliografia

[1] Grigoris Antoniou and Frank Van Harmelen. *A Semantic Web Primer*. MIT Press, 2004.

[2] Yukun Chen, Yu Zheng, Xing Xie, Kai Jiang, Chunping Li, and Nenghai Yu. Trajectory simplification method for location-based social networking services. In *SIGSPATIAL GIS workshop on location-based social networks*. Association for Computing Machinery, Inc., November 2009.

[3] Michele Fiorini, Andrea Capata, and Domenico D. Bloisi. Ais data visualization for maritime spatial planning (msp). *International Journal of e-Navigation and Maritime Economy*, 5:45 – 60, 2016.

[4] T. Warren Liao. Clustering of time series data—a survey. *Pattern Recognition*, 38(11):1857 – 1874, 2005.

[5] F. Mazzarella, M. Vespe, D. Damalas, and G. Osio. Discovering vessel activities at sea using ais data: Mapping of fishing footprints. In *17th International Conference on Information Fusion (FUSION)*, pages 1–7, 2014.

[6] Oracle. Oracle Spatial and Graph. http://www.oracle.com/technetwork/database/options/spatialandgraph/documentation/spatial-doc-idx-161760.html, 2018. Visitato il 26.04.2018.

[7] Oracle. Spatial Features Technical Information. http://www.oracle.com/technetwork/database/options/spatialandgraph/documentation/spatial-techinfo-152816.html, 2018. Visitato il 26.04.2018.

[8] Giuliana Pallotta, Michele Vespe, and Karna Bryan. Vessel pattern knowledge discovery from ais data: A framework for anomaly detection and route prediction. *Entropy*, 15(6):2218–2245, 2013.

[9] Heather M Perez, Roger Chang, Richard Billings, and Theodore L Kosub. Automatic identification systems (ais) data use in marine vessel emission estimation. In *18th Annual International Emission Inventory Conference*, volume 14, page e17, 2009.

[10] M. Potamias, K. Patroumpas, and T. Sellis. Sampling trajectory streams with spatiotemporal criteria. In *18th International Conference on Scientific and Statistical Database Management (SSDBM'06)*, pages 275–284, 2006.

[11] B. Ristic, B. La Scala, M. Morelande, and N. Gordon. Statistical analysis of motion patterns in ais data: Anomaly detection and motion prediction. In *2008 11th International Conference on Information Fusion*, pages 1–7, 2008.

[12] Alan Saalfeld. Topologically consistent line simplification with the douglas-peucker algorithm. *Cartography and Geographic Information Science*, 26(1):7–18, 1999.

[13] Guan Yuan, Penghui Sun, Jie Zhao, Daxing Li, and Canwei Wang. A review of moving object trajectory clustering algorithms. *Artificial Intelligence Review*, 47(1):123–144, Jan 2017.

[14] Yu Zheng. *Location-Based Social Networks: Users*, pages 243–276. Springer New York, New York, NY, 2011.

[15] Yu Zheng. Trajectory data mining: An overview. *ACM TIST*, 6(3):29:1–29:41, 2015.

[16] Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. Mining interesting locations and travel sequences from gps trajectories. In *Proceedings of*

the 18th International Conference on World Wide Web, WWW '09, pages 791–800, New York, NY, USA, 2009. ACM.

[17] Changqing Zhou, Dan Frankowski, Pamela Ludford, Shashi Shekhar, and Loren Terveen. Discovering personal gazetteers: An interactive clustering approach. In *Proceedings of the 12th Annual ACM International Workshop on Geographic Information Systems*, GIS '04, pages 266–273, New York, NY, USA, 2004. ACM.