Un'analisi sul servizio di Ferrovie Dello Stato

Silvia Bordogna 736610 - Federico Viotti 785867

Abstract— **Come espresso dalla dottrina sugli Opena Data, i dati relativi a servizi pubblici dovrebbero essere sempre piú liberamente accessibili a tutti in modo da favorire trasparenza e possibilitá di partecipazione al processo decisionale e alla valutazione del servizio stesso. In questo progetto si é voluto esplorare il mondo dei dati relativo al trasporto ferroviario italiano, cercando di valutare la qualitá del servizio espresso in termini di puntualitá dei treni e osservando se ci fosse un tentativo di comunicazione da parte dei fruitori del servizio tramite social network, ed eventuale conivolgimento da parte degli esercenti.**

I. INTRODUZIONE

Il seguente progetto nasce inizialmente con l'intenzione di monitorare orari effettivi del servizio ferroviario in concomitanza con l'attivitá su di un social network durante una giornata di sciopero. Dal momento che nel periodo di lavoro non si sono verificati scioperi nazionali, il processo realizzato é stato in veritá messo in produzione in una giornata qualunque, con conseguente impatto negativo sul livello di attivitá social che ci si attendeva in fase di ideazione. Ció nonostante il progetto ci é sembrato un discreto spunto per ipotetici sviluppi futuri.

II. SCRAPING SERVIZIO FERROVIARIO

Attualmente non sono disponibili dati aperti sullo storico del servizio effettivo prestato da Trenord per quanto riguarda la puntualitá dei treni. Per questo motivo si é reso necessario, per raggiungere lo scopo prefissato, realizzare uno scraper che sfrutta le informazioni puntuali circa lo stato di un treno e le raccoglie per poter creare una serie storica degli orari effettivi dei treni. Il sito utilizzato per raccogliere le informazioni é il seguente www.mobile.viaggiatreno.it, un dominio di Trenitalia usato per monitorare tratte e treni alla base di servizi quali www.viaggiatreno.it utilizzati dai cittadini per conoscere gli orari ferroviari su suolo nazionale.

A. Creazione lista treni

Prima ancora di poter raccogliere le informazioni sul funzionamento effettivo delle linee ferroviarie, si é dovuto creare una lista dei codici identificativi di ciascun treno, parametro necessario da inserire nella url di mobile.viaggiatreno.it per ottenere la lista delle fermate effettuate dal treno stesso con relativi orari effettivi e schedulati. Ad esempio per ottenere le informazioni relative al servizio regionale Malpensa Express su convoglio 365 é necessario inserire tale numero identificativo come parametro della url come segue

&numeroTreno=356&tipoRicerca=numero&lang=IT.

Dal momento che la lista dei numeri identificativi dei treni non é resa pubblica, si é adottato un metodo brute force per ottenerla. Si sono infatti provati tutti i numeri da 0 a 50000 durante l'arco di un intera giornata e, qualora il risultato non fosse nullo, si é inserito il valore in una lista dei treni, con relativa stazione di partenza e arrivo.

Dallo stesso sito, tramite una chiamata a

http://www.viaggiatreno.it/viaggiatrenonew/resteasy/viaggiatreno/autocompletaStazione

si é in seguito esportata la lista delle stazioni per ogni lettera dell'alfabeto di modo da poter confrontare se lista dei treni stesse coprendo un numero di stazioni rappresentativo. Infine, la lista validata dei treni é stata esportata come file csv.

B. Scraping orari effettivi

Una volta ottenuta la lista dei treni di interesse, la stessa url mostrata al punto precedente é stata utilizzata per trovare e salvare lo storico relativo al funzionamento effettivo dei servizi Trenitalia di un intera giornata.

Sebbene l'intenzione originale del progetto fosse quella di monitorare una giornata di sciopero, non é stato possibile in quanto non si sono verificati simili eventi entro la data di fine del progetto stesso. Per questo si é deciso di monitorare a titolo esemplificativo una qualunque giornata infrasettimanale, pur rimanendo consapevoli della minore rilevanza dell'impatto sui social network, che come si spiega nel punto successivo, costituisce la seconda fase del lavoro svolto.

Quindi per 24 ore la lista dei treni ricavati al punto precedente é stata ripetutamente iterata; ad ogni chiamata si ottiene un dato destrutturato, ovvero la pagina html che riporta lo stato effettivo del treno. Dal codice html, tramite regex, si sono estrapolate le sole informazioni di nostro interesse ovvero stazione, orario previsto e orario effettivo. Si mostra a titolo esemplificativo il risultato finale ottenuto dallo scraping di un singolo treno:

('MALPENSA AEROPORTO T2', '15:50', '15:50'),
('MALPENSA AEROPORTO T1', '15:55', '15:55'),
('BUSTO ARSIZIO', '16:03', '16:03'),
('SARONNO', '16:14', '16:14'),
('MILANO BOVISA', '16:26', '16:26'),

('M N CADORNA', '16:33', '16:32')

Alla fine del processo gli elementi sono stati unificati in un pandas dataframe.

In questa fase del processo sarebbe stato opportuno implementare un sistema di calcolo parallelo, dal momento che il tipo di operazione si presta ad essere divisa in blocchi di calcolo simultanei. Non avendo implementato questa soluzione ne risulta che lo scraping non copre la totalitá delle tratte. Il tempo infatti che trascorre tra lo scraping del primo e quello dell'ultimo treno é di circa un'ora, questo significa che qualora vi fossero treni che in quest'arco di tempo iniziano e concludono una nuova tratta, lo scraper non sarebbe in grado di recuperarne le informazioni.

Per stimare l'entitá della perdita si é usato il numero di corse giornaliere dichiarato sul sito di Trenord per quanto riguarda la regione Lombardia, da cui si deduce che il campione da noi trattato corrisponde a circa il 70% del totale delle corse.

C. Integrazione Regione Istat

Per arricchire l'analisi relativa al servizio si é quindi scelto di identificare la regione di ogni stazione. A tal fine si é utilizzato il dato reso disponibile dall'Istituto nazionale di statistica al segunete indirizzo www.istat.it/it/archivio/6789. Una volta effettuato il download si é provveduto ad integrare con il risultato dello scraping. Dal momento che la naming convention utilizzata da Trenitalia non corrispondeva a quella di Istat, si é reso necessario un lavoro di normalizzazione dei nomi dei comuni in tre step:

- Trasformazione in lettere maiuscole e rimozione accenti
- Sostituzione di tutti i "S." con "SAN" (ad esempio S. Donato Milanese diventa San Donato Milanese)
- Estrapolazione della prima parola in caso di nomi multipli (MILANO CENTRALE diventa MILANO)

Si é tentato successivamente di utilizzare la libreria Python difflib per trovare e aumentare la corrispondenza di nomi simili, senza peró miglioramenti significativi sul livello di record linkage raggiunto, motivo per cui il risultato é stato tralasciato.

III. Streaming del Social Network Twitter

I Social network sono canali che permettono di avere una connessione e un contatto digitale con il resto del mondo, ci permettono di condividere esperienze e di far conoscere noi stessi all' esterno. I Social sono anche una grande valvola di sfogo per mezzo della quale si possono esprimere opinioni positive e/o negative su qualunque tema o oggetto.

Come presentato nell' introduzione in questo progetto si é voluto sondare se e come consumatori e fornitori di un servizio, quello ferroviario, comunichino sul Social Network Twitter durante una giornata lavorativa.

A. API di Twitter

Twitter mette a disposizione una RESTful API che permette agli sviluppatori di integrare le proprie applicazioni con le funzioni del Social. E' stato quindi creato, ai fini del progetto, un account sviluppatore con piano gratuito (che permette di eseguire meno funzioni degli account a pagamento ma da' comunque la possibilitá di ottenere le informazioni necessarie). La API, che é integrabile con molte librerie python, fornisce inoltre la funzione di filtraggio dei tweet in tempo reale attraverso una o piú parole chiave.

B. Archiviazione RT dei tweet

1) Il dizionario: Per poter ottenere i tweet di un'intera giornata, relativi al servizio di trasporto ferroviario, si é proceduto da prima a creare una lista di parole chiave che lavorasse da dizionario per la API, cosí da permetere di filtrare i tweet scartando tutti quelli che non contenessero una o piú parole contenute in esso. Nel dizionario sono state inserite parole/frasi comuni (come *ferrovie dello stato*) e hashtag relativi ai fornitori principali del servizio ferroviario (come *#trenord* e *#trenitalia*).

2) Lo streaming e l'archiviazione: Successivamente alla creazione del dizionario si é proceduti alla realizzazione dello streaming dei tweet. Per fare ció la API é stata integrata in python tramite la libreria *Tweepy*.

Per prima cosa é stata creata una connessione tra lo script e una sessione del DBMS non relazionale MongoDB (é stato scelto MongoDB per il suo orientamento ai documenti, dato che il formato standard dei tweet ottenibili con la API é document based).

Per poter archiviare i tweet in tempo reale si é quindi generata una sessione di straming tramite un'istanza del comando *tweepy.stream*, che permette di connettersi ad un *Listener*, utile a mainpolare i tweet che vengono filtrati. All'interno del *Listener* é quindi stato creato un elenco in formato json che contenesse tutte le informazioni che dovevano essere trattenute dai tweet che venivano pubblicati. Ogni volta che un tweet veniva postato, questo veniva agganciato e archiviato in una collection di MongoDB.

Le informazioni ottenute dai tweet sono:

- nome dell'utente e alias
- location del profilo, non necessariamente un luogo
- descrizione, una stringa con cui l'utente definisce il suo profilo
- numero di followers dell'utente
- numero di persone seguite dall'utente
- numero di like e numero di tweet fatti dall'utente
- data del profilo e data di pubblicazione del tweet (comprensiva di orario)
- geolocalizzazione, se attiva e se presente all'interno del tweet
- lingua
- testo del tweet

- risorsa del tweet, una stringa contenente il supporto tramite il quale é stato pubblicato il tweet
- hashtag presenti nel tweet

C. Preprocessing

Lo streaming dei tweet é stato fatto durante 24 ore, in contemporanea allo scraping degli orari previsti ed effettivi. I tweet ottenuti sono stati quindi esportati da Mongo in csv tramite il comando mongo export.

Ottenuto il csv questo é stato opportunamente modificato: per prima cosa sono state eliminate le colonne relative alle informazioni non utili ai fini della successiva analisi; é stato successivamente cambiato il formato dell'ora (che presentava uno scarto di 2 ore indietro essendo il campo *created_at* basato sull'orario UTC); si é infine modificato il campo *risorsa tweet* aggregando i dati in 6 categorie:

- CorpService, supporti a servizio delle aziende
- Desktop, comune PC fisso
- Smartphone
- Twitter for Android
- Twitter for iPhone
- Uknown, ovvero supporti di cui non si é capita l'origine

IV. INTEGRAZIONE E ANALISI

Una volta ottenuti e preparati i dati, le due fonti sono state aggregate a dettaglio orario e integrate, creando un dataset finale contentente per ogni ora la somma dei ritardi accumulati dai treni e la somma dei tweet inerenti le keyword selezionate. Come detto in precedenza l'obiettivo principale dell'analisi era individuare una eventuale corrispondenza tra disservizio e lamentela su social network. Purtroppo i risultati ottenuti non sono molto soddisfacienti in quanto il numero di tweet é piuttosto basso. Inoltre non é stato possibile effettuare un match a livello di regione in quanto la maggior parte dei tweet non popola il campo location utente o dove questo é presente spesso risulta un luogo fittizio come ad esempio "Everywhere", "nord;non di nascita" e simili.

Si é realizzato un grafico che riporta il trend del numero di tweet di lamentela in concomitanza con la somma dei ritardi per ogni ora. Quello che possiamo dedurne é che il picco massimo di tweet si colloca nella fascia oraria tra le 7 e le 8. Al contrario, esclusa la fascia oraria notturna, il numero di interventi diminuisce in corrispondenza di fasce non di picco come quella tra le 10 e le 11 della mattina. Per quanto riguarda i ritardi, notiamo ovviamente un aumento del ritardo complessivo negli orari di punta, certamente dovuto all' incremento del servizio e dunque del ritardo complessivo.

Per questo si é analizzato con dettaglio l'andamento della mediana e del ritardo massimo come riportatao in Figura 2. Si nota che la mediana dei ritardi nella fascia di rientro a casa (17-19) non raggiunge mai picchi verso l'alto, ad indicare che nel momento di maggior affluenza, e con il maggior numero di corse, pare esserci stato un servizio migliore che non nei momenti dove la rete ferrovaria é meno attiva.

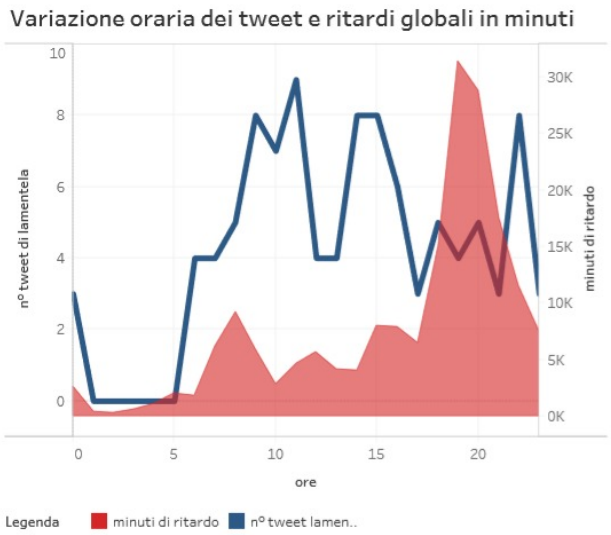

Fig. 1. Trend Ritardo Totale e Numero Tweet

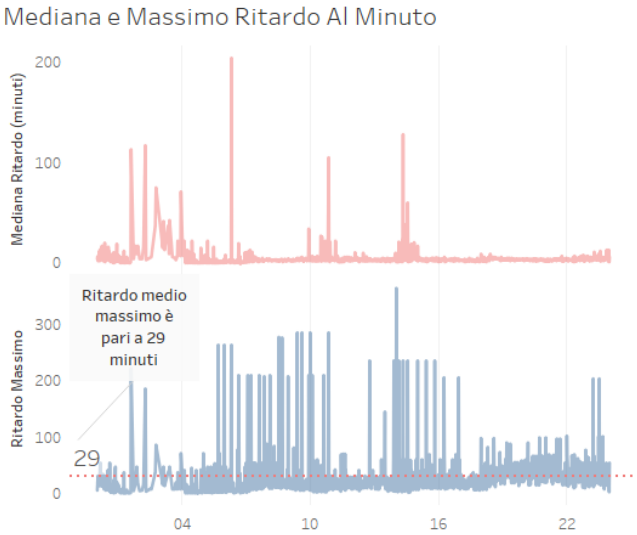

Fig. 2. Trend Mediana Ritardi per Minuto

Il dato aggregato nasconde peró le differenze regionali che, sia per la complessitá della rete che per il numero di passeggeri coinvolti, possono considerarsi degli ecosistemi fortemente diversi. Come emerge in Figura 3, il ritardo medio varia a seconda dell'area geografica. Si nota inoltre l'assenza di correlazione con la grandezza della rete di ogni regione, vi sono infatti regioni quali la Lombardia che pur detenendpo il 27% delle corse del campione in esame, si posiziona nella fascia piú bassa rispetto al ritardo medio. Al contrario in una regione come la Basilicata, il cui servizio corrisponde a circa 1% delle corse nazionali, nel giorno di raccolta dati almeno nel 50% delle fermate si é verificato un ritardo di 6 minuti.

Una diversa prospettiva viene data dal grafico in figura 4, dove sono stati aggregati i dati relativi alle regioni a formare le tre aree principali dello stato italiano (*nord, centro, sud*).

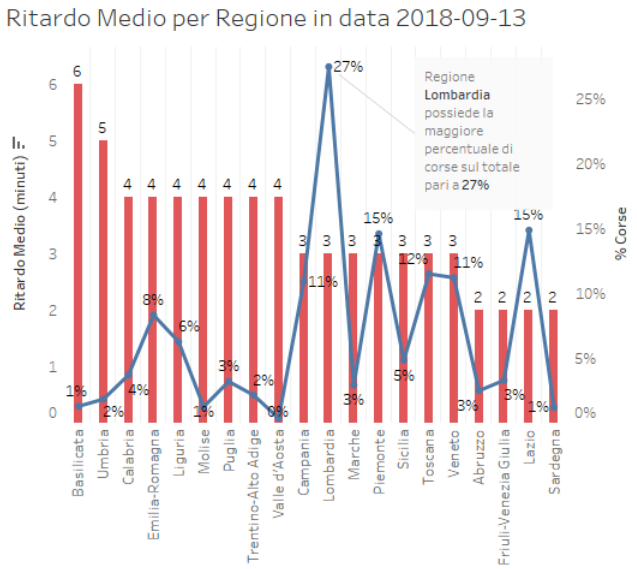

Fig. 3. Mediana Ritardo per Regione

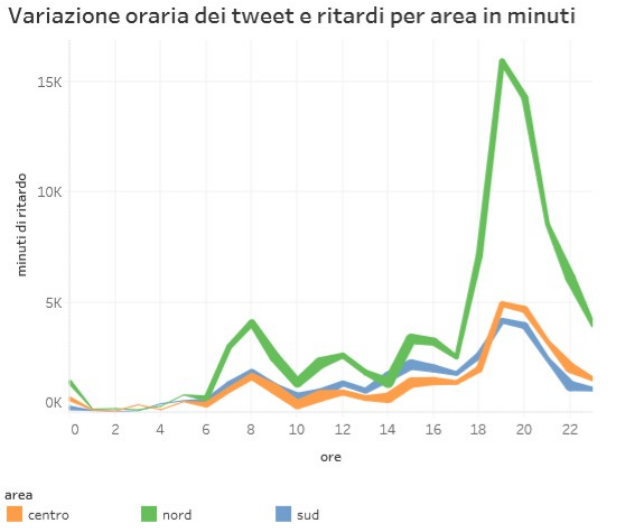

Fig. 4. Variazione numero tweet e somma ritardi per area geografica

Si nota come il Nord Italia presenti un ritardo complessivo dei treni maggiore alla somma dei ritardi del Centro e Sud Italia, detenendo peró circa il 60% delle corse effettuate nella giornata.

V. CONCLUSIONI

Dunque in questo progetto si sono seguite due delle tre V dei Big Data: varietá, data dalle diverse fonti (Twitter, www.mobile.viaggiatreno.it, Istat) e formati (json, html, csv). I dati, opportunamente integrati, non hanno mostrato grandissime evidenze anche se si puó comunque concludere che l'utilizzo di Twitter in relazione al servizio ferroviario cambia al variare degli orari e dei ritardi che si verificano durante una giornata lavorativa.

Possibili sviluppi futuri potrebbero consistere nel monitoring dei tweet e dei ritardi durante una giornata di sciopero del servizio (appunto l'originale intento di questo progetto) e mettere a confronto l'attivitá su Twitter con quella su Facebook per valutarne i diversi impatti e comportamenti dei pendolari.