



Emotional Maps

MANUALE TECNICO

a cura di

Luca Alberti, Simone Brenna, Nicolò Ferrari, Luigi Mucciarone



Questa pagina e stata lasciata bianca intenzionalmente.

Indice

COS'È EMOTIONAL MAPS.....	3
COMANDI SUPPORTATI.....	4
IMPORT	4
CREATE_MAP.....	5
STRUTTURE DATI UTILIZZATE	6
ARCHIVIAZIONE DEI DATI	6
DEFINIZIONE DEI VALORI DEGLI EVENTI	6
SCELTE IMPLEMENTATIVE	7
SCELTE TECNICHE.....	8
GESTIONE DEGLI ERRORI.....	9

COS'È EMOTIONAL MAPS

“Emotional Maps” è un software per l’elaborazione delle informazioni ottenute mediante riconoscimento facciale eseguito sulle foto scattate dagli utenti tramite “Facedoor” presso un POI e di estrarre una serie di dati statistici riguardanti gli stati emotivi registrati.

COMANDI SUPPORTATI

IMPORT

Il comando viene utilizzato per indicare il file contenente i file log delle Emotional Maps da importare.

La formattazione del comando è la seguente:

```
import(<percorso\nome_file>.txt)
```

Ad esempio, se scrivessimo:

```
import(C:\documenti\eventi1.txt)
```

il programma tenterà di accedere al file “eventi1.txt”. In caso di successo, gli eventi contenuti verranno caricati, invece se l’operazione ha riscontrato un qualsiasi tipo di errore, il programma proseguirà avvertendo l’utente del problema.

Il formato degli eventi contenuti nel file deve rispettare la seguente formattazione:

<stato_reg> <stato_utente> <data> <id_utente> <geo_coord> <stato_emotivo>

Sigla	Descrizione	Valori possibili / formato corretto
stato_reg	Indica lo stato di iscrizione dell’utente	IN/OUT
stato_utente	Indica lo stato di accesso dell’utente	LOGIN/LOGOUT
data	Indica la data dell’evento	La data deve essere conforme al seguente formato: GGMMAAAA
id_utente	Codice identificativo univoco dell’utente	Stringa alfanumerica di 5 caratteri.
geo_coord	Indicano in quale posizione geografica viene fatto il rilevamento	Due valori decimali separati da una virgola (,)
stato_emotivo	Indica lo stato emotivo dell’utente	A/F/S/T/N Rispettivamente: Arrabbiato, Felice, Sorpreso, Triste e Neutro

CREATE_MAP

Il comando richiede al programma di generare una mappa degli stati emotivi dei punti di interesse (POI) usando gli eventi presenti nell'intervallo di tempo selezionato. Ogni evento influenza la mappa emozionale del POI più vicino.

Ogni mappa emozionale viene visualizzata a schermo tramite una tabella indicando il punto di interesse con le percentuali delle emozioni divise in eventi attivi e eventi totali.

La formattazione del comando è la seguente:

```
create_map(<data_inizio>-<data_fine>)
```

Nota: La formattazione delle date deve essere GGMMAAAA.

Se ad esempio scrivessimo:

```
create_map(01042019-10042019)
```

Il programma elaborerà una serie di dati partendo dal 01/04/2019 fino al 10/04/2019 e al termine andrà a generare 2 output simili a questi:

POI1 – 47% A, 33% F, 0% S, 20% T, 10% N

POI2 – 15% A, 0% F, 40% S, 15% T, 30% N

POI3 – 10% A, 60% F, 10% S, 0% T, 20% N

realizzati, uno per i soli utenti online in quel momento e uno su tutti gli utenti.

Se il comando presenterà errori verrà generato a schermo un errore.

STRUTTURE DATI UTILIZZATE

ARCHIVIAZIONE DEI DATI

Per l'archiviazione degli eventi generati dagli utenti di Facedoor si è deciso di utilizzare un hashmap con chiave una stringa che rappresenta la data di generazione e come oggetto una lista dinamica contenente tutti gli eventi generati in quella data.

Questo ci permette di aggiornare e ricercare le informazioni sugli eventi con costo pari a $O(1)$.

Durante l'elaborazione delle mappe emozionali, il costo aumenta a $O(n)$, dove n sono le giornate presenti nell'intervallo di tempo usato per la generazione delle mappe emozionali.

I comandi convertiti vengono salvati all'interno di un Array di tipo Comando.

Il manager dei PDI è una Lista di tipo PuntoDiInteresse.

DEFINIZIONE DEI VALORI DEGLI EVENTI

I comandi e gli eventi sono rappresentati da stringhe alfanumeriche durante la loro archiviazione su file, per rendere le operazioni sui comandi e dati più rapide, si è deciso di usare le enumerazioni per la loro rappresentazione durante l'esecuzione del programma. Questo permette di poter gestire una mole di dati maggiore rispetto alla semplice rappresentazione tramite stringhe.

SCELTE IMPLEMENTATIVE

All'interno del codice sono stati utilizzati alcuni package standard di Java del tipo “.io” e “.util” per la gestione delle funzioni, tra le più importanti: lettura da file, gestione delle eccezioni, creazione di ArrayList, List e HashMap.

SCELTE TECNICHE

Se il file non presenterà alcuna estensione verrà aggiunta automaticamente l'estensione “.txt” attraverso un apposito controllo e non genererà alcun errore.

La correttezza dei comandi all'interno del file passato viene controllata tramite un match di regex.

Si è deciso di suddividere eventuali problemi in Errori e Avvisi. Gli Errori portano alla chiusura del programma mentre gli Avvisi fanno procedere il programma escludendo la parte di dati errata.

GESTIONE DEGLI ERRORI

Durante l'esecuzione del programma, si possono riscontrare Errori o AVVISI

Nel caso di ERRORE, il programma mostrerà a schermo la descrizione generale dell'errore per poi terminare. I possibili casi sono:

- Non è stato passato nessun argomento contenente i comandi;
- Il file dei comandi passato non esiste.

Nel caso di AVVISI, il programma proseguirà l'esecuzione e verranno mostrati a schermo degli avvisi. I casi possibili sono:

- Un comando non è stato riconosciuto dal programma perché scritto in modo errato oppure non è un comando designato;
- Il numero di argomenti di chiamata non è corretto;
- Il file contenente gli eventi da importare non esiste o è in un formato non supportato;
- Se il lasso di tempo indicato dal comando `create_map` non presentasse nessun evento;
- Se la sintassi del codice dell'evento non fosse conforme.

Nota: A seconda della natura dell'errore, i risultati finali potrebbero essere imprecisi.