

# POLITECNICO DI BARI



**DIPARTIMENTO DI INGEGNERIA ELETTRICA  
E DELL'INFORMAZIONE  
CORSO DI LAUREA IN INGEGNERIA INFORMATICA E  
DELL'AUTOMAZIONE**

---

**TESI DI LAUREA  
in  
SISTEMI OPERATIVI**

***Rappresentazione della Conoscenza e Ragionamento  
Automatico per l'Affective Computing a partire da  
Biosegnali***

**Relatore:**

Prof. Ing. Michele Ruta

**Correlatori:**

Ing. Floriano Scioscia

Ing. Giuseppe Loseto

Ing. Agnese Pinto

**Laureando:**

Nicolò Santilio

**Anno Accademico 2018/2019**



Politecnico  
di Bari

**LIBERATORIA ALLA CONSULTAZIONE DELLA TESI DI LAUREA DI CUI  
ALL'ART.4 DEL REGOLAMENTO DI ATENEO PER LA CONSULTAZIONE DELLE  
TESI DI LAUREA (D.R. n. 479 del 14/11/2016).**

Il sottoscritto Nicolò Santilio matricola 568745

Corso di Laurea  
INGEGNERIA INFORMATICA E DELL'AUTOMAZIONE (D.M.270/04)

autore della presente tesi di Laurea dal titolo  
Rappresentazione della Conoscenza e Ragionamento Automatico per l'Affective Computing  
a partire da Biosegnali

Parola chiave

Abstract

☒ Autorizza

☐ Non autorizza

la consultazione della presente tesi, fatto divieto a chiunque di riprodurre in tutto o in parte quanto  
in essa contenuto.

Bari,

Firma Nicolò Santilio



# Sommario

<b>1.</b>	<b>Introduzione</b>	<b>3</b>
<b>2.</b>	<b>Rappresentazione della conoscenza per l'Affective Computing</b>	<b>6</b>
2.1.	Affective Computing	6
2.1.1.	Modelli per la descrizione delle emozioni	7
2.1.2.	Segnali fisiologici per il riconoscimento delle emozioni	9
2.1.3.	Il dataset DECAF	12
2.2.	Rappresentazione della Conoscenza	15
2.2.1.	Logiche Descrittive	15
2.2.2.	Servizi di inferenza	17
2.3.	Ontologia per il riconoscimento delle emozioni	20
<b>3.</b>	<b>Riconoscimento delle emozioni mediante inferenza automatizzata</b>	<b>27</b>
3.1.	Descrizione del problema	27
3.2.	Generazione dell'ABox	27
3.3.	Matchmaking semantico per il riconoscimento delle emozioni	33
<b>4.</b>	<b>Esperimenti</b>	<b>37</b>
4.1.	Metodologia sperimentale	37
4.2.	Risultati	39
<b>5.</b>	<b>Conclusioni</b>	<b>47</b>
<b>6.</b>	<b>Bibliografia</b>	<b>49</b>
<b>7.</b>	<b>Ringraziamenti</b>	<b>51</b>

# Capitolo 1

## Introduzione

L'elaborazione e la comprensione da parte di sistemi automatici delle emozioni umane, oltre a costituire un passo avanti molto importante per l'intelligenza artificiale e l'interazione uomo-macchina, risulta essere la chiave per il miglioramento di tutti quegli ambiti applicativi in cui lo stato emotivo e cognitivo dell'individuo rappresenta un importante dato, al fine di migliorare il benessere dell'individuo stesso.

Molteplici sono gli aspetti che rendono complessi i processi di acquisizione, interpretazione ed elaborazione dei biosegnali utili ad identificare e riconoscere le emozioni umane, tra cui l'utilizzo di tecnologie pervasive e non invasive per catturare gli aspetti caratteristici della loro espressione (come la comunicazione paraverbale e non verbale), l'identificazione di algoritmi ottimali per la loro classificazione e l'utilizzo attraverso risposte rapide e coerenti di *biofeedback* per migliorare il benessere del soggetto nello stato emotivo identificato. Tutti questi elementi costituiscono oggetto di studio dell'*Affective Computing*, che si pone all'intersezione tra l'Intelligenza Artificiale, la Fisiologia e la Psicologia.

Utilizzato sia per la *sentiment analysis* [1] più puntuale (tecnica ormai ampiamente utilizzata nel mondo del business, comunicazione e marketing), che per il trattamento di situazioni patologiche o di alterazioni nello stato psico-fisico del paziente, il rilevamento automatizzato delle emozioni richiede l'estrazione di caratteristiche (*feature*) specifiche, spesso memorizzate all'interno di dataset molto voluminosi e complessi, con la conseguenza di costi computazionali elevati o limitazioni in termini operativi di elaborazione dei dati.

Inoltre, è spesso necessario effettuare minuziose operazioni preliminari di pulizia e filtraggio dei dati ottenuti da differenti biosensori, affinché le tecniche di apprendimento automatico (*machine learning*), necessarie al processing delle emozioni, offrano risultati attendibili.

Nonostante lo sviluppo e la diffusione di sensori indossabili (*wearable*) a basso costo ha reso possibile una più facile acquisizione dei biosegnali, permettendo la creazione di architetture sensoriali più accessibili, uno dei nodi cruciali relativi all'*Affective Computing* risiede ancora nell'utilizzo di software specifici per piattaforme di calcolo tradizionali – quali desktop o laptop – per l'interpretazione e la classificazione dei dati per il riconoscimento delle emozioni. Attualmente le tecniche principalmente adoperate per affrontare queste analisi consistono in algoritmi di machine learning, sistemi basati su regole e ragionamento deduttivo.

Quest'ultimo, nello specifico, è un approccio promettente per migliorare l'affidabilità e la qualità dell'output ottenuto, in quanto permette di ottenere risposte che non si limitino ad etichettare semplicemente l'emozione rilevata, ma che risultino esaustive ed interpretabili, essendo in grado di evidenziare tutte le possibili relazioni tra l'esito del processo di identificazione e le osservazioni effettuate. Sui risultati così ottenuti è possibile applicare ulteriori servizi di inferenza standard e non-standard [2] [20].

In particolare, i servizi di inferenza si basano su strumenti per la *Rappresentazione della Conoscenza*, branca dell'intelligenza artificiale il cui scopo è quello di studiare e formalizzare il ragionamento umano attraverso linguaggi in grado di rappresentare l'informazione in modo formale e non ambiguo, col fine di renderla comprensibile alle macchine. L'inferenza è dunque il metodo attraverso cui è possibile dedurre conoscenza implicita proveniente da una *ontologia* specifica, che concettualizza la conoscenza relativa ad un certo *dominio* in maniera esplicita.

I linguaggi adoperati dalla Rappresentazione della Conoscenza, comprensibili dunque tanto dall'essere umano che da un elaboratore, forniscono tutti gli strumenti necessari per definire regole sulle quali costruire asserzioni sulla base di conoscenze, dei quantificatori per darne un significato, e dei valori di verità rispetto ad un modello. La potenzialità di questi linguaggi si esprime prevalentemente in due caratteristiche fondamentali:

- la possibilità di definire linguaggi sufficientemente espressivi da permettere la descrizione di un dominio di interesse specifico;
- la possibilità di definire linguaggi dotati di una complessità ed espressività tale da richiedere quantità accettabili di risorse di calcolo necessarie per essere adoperati in procedure di inferenza nell'ambito di uno specifico scenario applicativo di riferimento.

Attraverso un approccio basato sulla rappresentazione della conoscenza, è possibile costruire un framework di riconoscimento delle emozioni che utilizzi una ontologia in grado di descrivere ciascuna emozione a partire da un dataset prestabilito di rilevazioni di biosegnali, sfruttando gli strumenti annotazione dei dati ed inferenza automatizzata forniti da un ragionatore espressamente sviluppato per dispositivi embedded [2].

Il lavoro di tesi intende studiare e sperimentare un possibile approccio alla creazione di un framework di riconoscimento delle emozioni mediante ragionamento automatizzato, sfruttando una base di conoscenza basata su biosegnali ricavati da prove di laboratorio appositamente effettuate per la rappresentazione delle emozioni.

In particolare, per il raggiungimento di tale scopo è stato progettato e implementato un workflow di operazioni che hanno messo in evidenza l'efficienza di elaborazione di tale sistema e le potenzialità esprimibili attraverso processi di raffinamento dei dati. Tali potenzialità sono rappresentate da:

- possibilità di formalizzare basi di conoscenze a partire da dati provenienti dalla misurazione di biosegnali, attraverso strumenti comprensibili e scalabili;
- esecuzione di complesse operazioni di ragionamento su base semantica, attraverso il ragionatore *Mini-ME (Mini Matchmaking Engine)* [2] sviluppato dal Laboratorio di Sistemi Informativi del Politecnico di Bari, con basso dispendio di risorse computazionali e ridotti tempi di esecuzione.

## Capitolo 2

### Rappresentazione della conoscenza per l'Affective Computing

#### 2.1 Affective Computing

È ormai consolidata la letteratura scientifica che mostra come le emozioni giochino un ruolo essenziale nel processo decisionale, nella percezione, nell'apprendimento e in molti altri ambiti dell'agire umano.

Come afferma Rosalind Picard [4], se desideriamo che i computer siano davvero intelligenti e che riescano ad interagire in modo naturale con l'uomo, è necessario che essi posseggano tutti gli strumenti per riconoscere, comprendere ed “esprimere” delle emozioni.

Oramai divenuto di grande interesse negli ultimi anni, l'*Affective Computing* risulta essere un campo interdisciplinare vastissimo, il cui obiettivo è quello di creare sistemi artificiali con abilità *empatiche*, capaci cioè di interpretare lo stato emotivo dell'uomo attraverso il linguaggio verbale (la comunicazione diretta, orale o scritta), non verbale [12] (l'insieme dei parametri fisiologici ottenibili, dall'espressione facciale ai biosegnali) e paraverbale [13] (la tonalità e il carico emotivo del parlato).

L'emozione in sé è considerata una variabile fondamentale nell'equazione di ciò che attualmente definiamo intelligenza [5] [6]: una delle sue tante forme, detta *intelligenza sociale* o *intelligenza emotiva*, risulta essere strettamente legata all'empatia, poiché richiede l'abilità di riconoscere ed esprimere le emozioni, ma soprattutto far esperienza delle emozioni altrui.

Il potenziale contributo dell'Affective Computing non è solo di tipo teorico, ma anche pratico: non solo, infatti, permette di approfondire gli studi condotti dalle neuroscienze, dalla psicologia e dalle scienze cognitive nel complesso ambito della percezione delle emozioni (termine sul quale ancora oggi si studia per darne una definizione certa), ma soprattutto fornisce alle macchine un approccio “emotivo” nella risoluzione di problemi di natura umana e logica, evolvendo le possibilità comunicative tra uomo e macchina e aprendo nuove strade per lo sviluppo umano stesso a livello individuale e sociale.

Tra i principali campi di applicazione dell'Affective Computing è possibile considerare:

- assistenza sanitaria, con l'utilizzo di robot sociali per sopperire alla progressiva carenza di personale dedito alla gestione della popolazione anziana, ma anche con l'applicazione e sviluppo di tecnologie utilizzabili dalle persone affette da autismo nello sviluppo delle capacità comunicative [7];



- istruzione, per il miglioramento dell'apprendimento da parte degli studenti mediante strumenti di riconoscimento del livello di apprendimento attraverso l'umore e lo stato emotivo, e l'eventuale attuazione di piani di insegnamento adattivi;
- marketing, con l'instaurazione di un doppio canale di comunicazione attraverso il quale un'azienda è in grado di recepire come un cliente percepisce a livello emotivo un prodotto, e di conseguenza migliorare la comunicazione del prodotto stesso ottenendo l'effetto desiderato.

### 2.1.1 Modelli per la descrizione delle emozioni

Per comprendere *come* un calcolatore possa essere in grado di riconoscere le emozioni, è innanzitutto fondamentale comprendere *da che cosa* un'emozione può essere rappresentata in ambito digitale.

Come già affermato, l'espressione di un'emozione umana si presenta sotto diverse forme, implicite ed esplicite, attraverso il complesso meccanismo della comunicazione. Sia dalle numerose ricerche scientifiche, sia dalla quotidiana esperienza umana, risulta evidente che le emozioni influenzano in maniera preponderante sia la psicologia che la fisiologia di ogni individuo.

Nonostante i notevoli sviluppi nell'analisi del linguaggio naturale attraverso tecniche di intelligenza artificiale sempre più efficaci nel riconoscere i tratti particolari della comunicazione emotiva, risulta ancora difficoltosa una corretta analisi dei biosegnali in relazione ad una reazione fisiologica scaturita da una emozione. È infatti importante ottenere dati non influenzati dalla sfera sociale, misurabili in tempo reale e soprattutto non modificati da artefatti di natura strutturale (precisione dei sensori utilizzati) ed individuale (la diversa fisiologia da individuo ad individuo).

Analizzando le risorse attualmente in letteratura, è stato individuato il dataset DECAF [8], punto di partenza dell'approccio proposto nel lavoro di tesi. DECAF presenta una serie di rilevazioni attraverso cui è possibile individuare e catturare la reazione fisiologica ad un determinato stimolo, intento a suscitare una emozione pilotata in un paziente. La risposta fisiologica è rilevata attraverso l'utilizzo di appositi sensori in grado di registrare i biosegnali emessi da un individuo, relativi a risposte emotive esplicite ed implicite.

I biosegnali, però, si presentano sotto forma di flussi di dati numerici, che è dunque necessario contestualizzare nell'ambito dell'informazione affettiva, attraverso dei modelli in grado di descrivere le singole emozioni.

Tra i diversi modelli supportati nella ricerca dell'Affective Computing, ve ne sono due fondamentali, utilizzabili per l'analisi del dataset considerato: il modello *categoriale* [9] ed il modello *VAD* [10].

Per il modello categoriale, ad opera di P. Ekman [9], le esperienze emotive possono essere raccolte all'interno di un gruppo ristretto di *emozioni universali*, cioè presenti in ogni contesto geografico e culturale, costituito da: *rabbia*, *disgusto*, *paura*, *felicità*, *tristezza* e *sorpresa*. Nonostante Ekman abbia successivamente ampliato il gruppo di emozioni universali, questo tipo di modello rimane il più utilizzato per il riconoscimento automatizzato delle emozioni, racchiudendole in una cerchia più ristretta poiché considerata come biologicamente appartenente alla stessa origine di esperienza. Ciò permette di identificare ciascuna emozione con un insieme di serie di biosegnali preliminarmente registrati, e valutare la corrispondenza con l'emozione che si intende suscitare con gli stimoli forniti. Un esempio applicativo del modello categoriale si trova nel sistema FACS (*Facial Action Coding System*) ad opera dello stesso Ekman [24].

Nella creazione di un dataset di riferimento, occorre tenere presente che l'emozione *effettiva* ottenuta come reazione alla percezione di uno stimolo può risultare distante o persino completamente differente rispetto all'emozione *ritenuta valida* per quello stesso stimolo. Ciò può dipendere da una serie di fattori, che variano da come le emozioni possono essere espresse in contesti controllati come quelli di un laboratorio, fino a più complessi meccanismi soggettivi di ogni individuo. È dunque necessario trovare un modello attraverso il quale una emozione possa essere descritta senza dipendere da una rappresentazione universale, ma da una più personalizzata. Questa visione è alla base del modello *Valence – Arousal – Dominance* (VAD) [11], teorizzato e proposto da M. Bradley nel 1994. Esso ritiene che l'intera gamma delle emozioni umane possa essere rappresentata attraverso un grafico tridimensionale di tipo valence-arousal-dominance. In particolare, il grafico VAD parte da un modello bidimensionale valence-arousal, dove per *valence* si intende la polarità dell'emozione (valutata come *piacevole* o *spiacevole*), ovvero come essa viene percepita soggettivamente dall'individuo oggetto dello stimolo; per *arousal* si intende invece l'intensità di tale emozione in termini di risposta allo stimolo stesso (valutata come *calma* o *eccitante*). In aggiunta a queste due è presente anche la *dominance*, che esprime l'entità del controllo esercitato da un determinato stimolo su chi lo percepisce, in una scala tra forte/dominante a debole/remissivo.

Nel modello VAD per accertare la corrispondenza tra emozione *effettiva* ed emozione *premeditata* gli stimoli da utilizzare siano scelti sulla base di una coppia di verità valence-arousal ottenuta da una più

grande popolazione di esperienze di emozioni preliminari. Un esempio di schema grafico VA: VAD può essere osservato in Figura 1 [12].

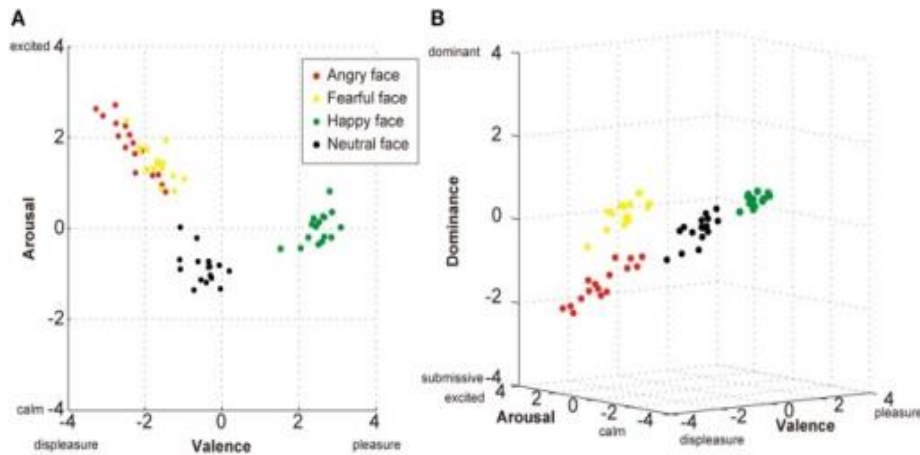


Figura 1: schema grafico bidimensionale VA (A) e tridimensionale VAD (B)

### 2.1.2 Segnali fisiologici per il riconoscimento delle emozioni

Sono trattati di seguito i biosegnali che caratterizzano il dataset DECAF [8] descrivendone in dettaglio il substrato fisiologico, le modalità di acquisizione adottate nell'approccio studiato e le feature che li caratterizzano.

Innanzitutto occorre precisare alcune definizioni introduttive:

- *Biosegnale*: segnale di natura chimica, elettrica o meccanica rilevato negli esseri viventi. I biosegnali possono essere rilevati attraverso molteplici tecniche, da cui è possibile estrapolare informazioni utili a livello clinico e fisiologico;
- *Feature*: singola caratteristica di osservazione relativa ad un biosegnale, può coincidere con un singolo organo o una parte di esso osservato durante la sua attività, che assume uno o più valori numerici che lo identificano.

Il framework proposto per il riconoscimento delle emozioni è basato sui biosegnali rilevati ed analizzati nel dataset DECAF, un database multidimensionale per la decodifica delle risposte emotive dei pazienti, rilevati attraverso magnetoencefalografia [13] [14] [15]. I biosegnali sono stati rilevati sottoponendo 30 partecipanti (16 uomini e 14 donne) alla visione di 36 clip video provenienti da diversi tipi di contenuti multimediali (ad esempio film di Hollywood o video musicali) [16]. A ciascuna clip è stata associata una particolare emozione, determinata attraverso studi statistici preliminari sulla base di un gruppo molto più ampio di contenuti multimediali, selezionando solo le risorse sicuramente in grado di evocare una determinata emozione.

Queste emozioni, corrispondenti a quelle fondamentali ottenute attraverso il modello categoriale, sono: *amusing, funny, happy, exciting, angry, disgusting, fear, sad, shock*. Inoltre, come mostrato in Tabella 1, per ciascuna delle clip sono stati annotati i valori, riferiti dai soggetti partecipanti, di valence e arousal per la rappresentazione attraverso il modello VAD e sono stati calcolati i parametri statistici di media  $\mu$  e deviazione standard  $\sigma$  attraverso i quali è stato possibile estrarre le clip più influenti dal punto di vista emotivo.

Emozione	ID	Film sorgente	L	Valence		Arousal	
				$\mu$	$\sigma$	$\mu$	$\sigma$
Amusing	01	Ace-Ventura: Pet Detective	102.1	1.22	0.53	1.03	1.00
	02	The Gods Must be Crazy II	67.1	1.56	0.50	1.20	0.96
	04	Airplane	85.2	0.99	0.83	1.15	0.88
	05	When Harry Met Sally	100.2	1.05	0.61	1.08	1.02
	**	Modern Times	106.4	0.86	0.69	-0.35	0.86

Tabella 1: descrizione di una sotto parte della Tabella 1 in [8]

I biosegnali utilizzati appartenenti al database multidimensionale sono i seguenti:

1. Magnetoencefalografia (MEG): è una tecnica diagnostica medica non invasiva che consiste nella registrazione dell'attività cerebrale basata su sensori SQUID (Superconducting QUantum Interference Device), che consentono di rilevare campi magnetici molto piccoli. Oltre ad essere più vantaggiosa rispetto all'elettroencefalografia (EEG) dal punto di vista prestazionale, la MEG consente di rilevare segnali provenienti da specifiche zone del cervello tollerando anche piccoli movimenti da parte dei pazienti durante la registrazione.
2. Elettrocardiografia (ECG): largamente utilizzata nel riconoscimento delle emozioni, sfrutta l'utilizzo di tre sensori applicati al paziente: due elettrodi posti sul polso ed uno di riferimento posto sul cubito permettono di rilevare i battiti cardiaci [15] [16];
3. Elettrooculografia (hEOG): tecnica utilizzata per la misurazione dei movimenti dell'occhio, la fissazione e il batter d'occhio. In particolari, ai fini del riconoscimento delle emozioni, è di interesse il rilevamento dei movimenti orizzontali dell'occhio, misurato ponendo due elettrodi sul lato destro e sinistro del viso del paziente. In questa maniera è stato anche possibile rilevare i movimenti zigomatici.
4. Elettromiografia trapezoidea (tEMG): metodo finalizzato al rilevamento dell'attività muscolare del paziente, attraverso la registrazione degli impulsi elettrici ad essa legati. Facendo esperienza di una emozione si può assistere all'espressione di diversi movimenti muscolari. Molti di questi sono involontari, originati principalmente da uno stato alterato (ansia, nervosismo, stress emotivo [18]). Attraverso l'EMG trapezoideo, ponendo dunque elettrodi

bipolari sul muscolo trapezio, è stato possibile legare tali attività involontarie con il livello di stress del paziente.

Ciascuno dei biosegnali rilevati possiede una serie di feature specifiche parametrizzabili attraverso un range di valori ben delineato. le feature rilevate sono descritte di seguito.

- Per la MEG è stato scansionato un range di frequenza 1-45Hz, filtrato tramite gradiometri e magnetometri e suddiviso in nove differenti canali in base alle diverse regioni del cervello, ottenendo le feature
  - *vertex*,
  - *left temporal* (lobo temporale sinistro),
  - *right temporal* (lobo temporale destro),
  - *left parietal* (lobo parietale sinistro),
  - *right parietal* (lobo parietale destro),
  - *left occipital* (lobo occipitale destro),
  - *right occipital* (lobo occipitale destro),
  - *left frontal* (lobo frontale destro),
  - *right frontal* (lobo frontale destro).
- Per l'ECG sono state estratte informazioni dal segnale originario ed è stata analizzata la densità spettrale di potenza:
  - *Heart beats (HB)*: rilevati attraverso l'onda R e la rimozione delle componenti a bassa frequenza, computando intervalli tra battiti (*Inter-Beat Intervals*), frequenza cardiaca (*Heart Rate*) e la variabilità della frequenza cardiaca (*Heart Rate Variability*);
  - *Power Spectral Density (PSD)*: estraendo una frequenza di range 0-5 Hz all'interno della frequenza di campionamento dell'ECG di 1 KHz. Questo range è stato poi suddiviso in 11 sottointervalli differenti, dei quali 10 contenenti solo i valori medi di ampiezza, 4 di *overband* ottenuti da un sottocampionamento alle basse frequenze, e 10 di *subband* ottenuti come valori logaritmici di PSD suddividendo l'intervallo di frequenze [0, 2.4].
- Per l'hEOG sono stati analizzati sia i movimenti orizzontali degli occhi, sia l'attività della muscolatura facciale:
  - *Eye movements*: estraendo le feature fondamentali attraverso un filtraggio passa-basso del segnale a 5 Hz, è stata calcolata la PSD suddividendo la banda tra 0 e 2 Hz in 8 sottointervalli, corrispondenti a 8 feature fondamentali per descrivere il movimento dell'occhio;

- *Facial muscle activity*: relativamente ai movimenti zigomatici più evidenti, filtrando il segnale nella banda 105-145 Hz e rilevando i valori di PSD mediata nel tempo con frequenza 1 Hz e suddivisa in 3 sottointervalli. In questo modo è stato ottenuto un totale di 11 features dalla hEOG.
- Per la tEMG gli elettrodi utilizzati hanno permesso di rilevare, in un range 45-145 Hz, le attività cardiache e le attività muscolari:
  - *Heart activities*: a seguito di un filtraggio passa-basso è stata calcolata la PSD su 9 intervalli di frequenza, che descrivono le attività cardiache in assenza di ECG;
  - *Muscle activities*: è stata la PSD media nelle bande [55,95] Hz e [105,145] Hz con frequenza 1 Hz per identificare le attività del muscolo trapezio

### 2.1.3 Analisi del dataset DECAF

Il dataset DECAF è stato pubblicato<sup>1</sup> dall'Università di Trento e reso liberamente accessibile – previa richiesta – a fini di ricerca. Esaminati nel paragrafo precedente i biosegnali utilizzati per il riconoscimento delle emozioni e le loro feature estratte, ora si analizza come è strutturato il dataset a livello pratico e in che maniera le feature sono state utilizzate.

DECAF è un database multidimensionale in formato MATLAB costituito da un insieme di tabelle complesse in forma matriciale di dimensioni  $A \times B \times C$ , ciascuna delle quali rappresenta un biosegnale acquisito dall'insieme dei soggetti partecipanti agli esperimenti nella visione di una determinata clip. Per comodità d'esposizione, si assumono le seguenti notazioni per la descrizione delle strutture dati e delle loro componenti seguendo le indicazioni ufficiali<sup>2</sup>:

- NS: numero di soggetti, nel caso di studio 30;
- NV: numero di clip video, nella fattispecie 36;
- NR: numero di gruppi dei canali MEG, nel caso specifico 9.

Tutte le matrici rispettano un ordine ben preciso, definito da una matrice bidimensionale aggiuntiva chiamata “PermutationList” di dimensioni  $NS \times NV$ , dove ciascuna riga contiene in ordine casuale l'ID di ogni video (da 1 a 36) visionato da un paziente.

<sup>1</sup> Sito Web DECAF Dataset: <http://mhug.disi.unitn.it/wp-content/DECAF/DECAF.html>

<sup>2</sup> <http://mhug.disi.unitn.it/wp-content/DECAF/DECAF/decaf-doc-CLS-TCR.pdf>

Si possono ora descrivere le matrici contenenti le feature di ogni segnale:

1. **ECG Features:** una matrice tridimensionale di dimensioni  $NS \times NV \times 92$ , dove 92 rappresenta il numero totale delle caratteristiche impiegate per la classificazione (Tabella 2);
2. **EMG Features:** una matrice tridimensionale di dimensioni  $NS \times NV \times 66$ , dove 66 rappresenta il numero totale di feature di tEMG (Tabella 3);
3. **EOG Features:** una matrice tridimensionale di dimensioni  $NS \times NV \times 66$ , dove 66 rappresenta il numero totale di feature di hEOG (Tabella 4).

ECG Features
01-06: statistical measurements over Inter-Beat-Intervals
07-12: statistical measurements over Heart-Rate
13-18: statistical measurements over Heart-Rate-Variability
19-22: Welch power spectral density of the ECG signal over [0; 0.1), [0.1; 0.2), [0.2; 0.3), and [0.3; 0.4)
23-32: logarithm PSD obtained for the sub-bands obtained on dividing [0, 2.4] into 10 equal intervals
33-38: statistical measurements over power spectral density of the [0.1; 0.2) band of the ECG signal
39-44: statistical measurements over power spectral density of the [0.2; 0.3) band of the ECG signal
45-50: statistical measurements over power spectral density of the [0.3; 0.4) band of the ECG signal
51-56: statistical measurements over power spectral density of the [0.4; 0.5) band of the ECG signal
57-62: statistical measurements over power spectral density of the [0.5; 0.6) band of the ECG signal
63-68: statistical measurements over power spectral density of the [0.6; 1) band of the ECG signal
69-74: statistical measurements over power spectral density of the [1; 1.5) band of the ECG signal
75-80: statistical measurements over power spectral density of the [1.5; 2) band of the ECG signal
81-86: statistical measurements over power spectral density of the [2; 2.5) band of the ECG signal
87-92: statistical measurements over power spectral density of the [2.5; 5) band of the ECG signal
Statistical measurements mean, std, skewness, kurtosis, % values > mean+std, % values < mean-std

*Tabella 2: Feature relative al biosegnale EMG*

tEMG Features
01-06: statistical measurements over power spectral density of the [0; 0.5) band of the tEMG signal
07-12: statistical measurements over power spectral density of the [0.5; 1.5) band of the tEMG signal
13-18: statistical measurements over power spectral density of the [1.5; 2.5) band of the tEMG signal
19-24: statistical measurements over power spectral density of the [2.5; 3.5) band of the tEMG signal
25-30: statistical measurements over power spectral density of the [3.5; 5.0) band of the tEMG signal
31-36: statistical measurements over power spectral density of the [5.0; 10) band of the tEMG signal
37-42: statistical measurements over power spectral density of the [10; 15) band of the tEMG signal
43-48: statistical measurements over power spectral density of the [15; 25) band of the tEMG signal
49-54: statistical measurements over power spectral density of the [25; 45) band of the tEMG signal
55-60: statistical measurements over power spectral density of the [55; 95) band of the tEMG signal
61-66: statistical measurements over power spectral density of the [105; 145) band of the tEMG signal
Statistical measurements mean, std, skewness, kurtosis, % values > mean+std, % values < mean-std

*Tabella 3: Feature relative al biosegnale tEMG*

hEOG Features
01-06: statistical measurements over power spectral density of the [0; 0.1) band of the hEOG signal

07-12: statistical measurements over power spectral density of the [0.1; 0.2) band of the hEOG signal
13-18: statistical measurements over power spectral density of the [0.2; 0.3) band of the hEOG signal
19-24: statistical measurements over power spectral density of the [0.3; 0.4) band of the hEOG signal
25-30: statistical measurements over power spectral density of the [0.4; 0.6) band of the hEOG signal
31-36: statistical measurements over power spectral density of the [0.6; 1) band of the hEOG signal
37-42: statistical measurements over power spectral density of the [1; 1.5) band of the hEOG signal
43-48: statistical measurements over power spectral density of the [1.5; 2) band of the hEOG signal
49-54: statistical measurements over power spectral density of the [105; 115) band of the hEOG signal
55-60: statistical measurements over power spectral density of the [115; 130) band of the hEOG signal
61-66: statistical measurements over power spectral density of the [130; 145) band of the hEOG signal
Statistical measurements mean, std, skewness, kurtosis, % values > mean+std, % values < mean-std

Tabella 4: Feature relative al biosegnale hEOG

4. **MEG Features:** una matrice contenente due vettori di dimensioni NS, di cui uno corrisponde ai dati dei gradiometri e l'altro ai dati dei magnetometri. Ogni cella include una matrice tridimensionale delle dimensioni di  $216 \times NV \times NR$ , dove 216 ( $6 \times 6 \times 6$ ) è il numero totale di coefficienti DCT (*discrete cosine transform*, trasformata coseno discreta) estratti dai segnali MEG di uno specifico gruppo di canali in risposta a un video specifico (Figura 2).

Come già specificato in precedenza, per ridurre la dimensionalità dei dati preservando le informazioni spaziali, i 102 canali sono stati divisi in nove gruppi in base alla funzionalità delle diverse zone del cervello.

Per la classificazione in prova singola, da ciascun lobo del cervello sono stati utilizzati i primi  $n = 2$  coefficienti DCT dalle dimensioni spaziale, temporale e spettrale per ottenere un totale di  $9 \times 8 = 72$  caratteristiche.

In definitiva, dunque, risultano 8 ID corrispondenti alle 8 feature derivanti dal ridimensionamento della matrice tridimensionale MEG Features che da  $6 \times 6 \times 6$  coefficienti viene ridotta all'utilizzo di  $2 \times 2 \times 2$  coefficienti. È pertanto importante ricordare che, al momento della selezione delle feature, va eseguita una operazione di reshaping della matrice per estrarre i primi 2 coefficienti per ogni dimensione, come mostrato in Figura 2.

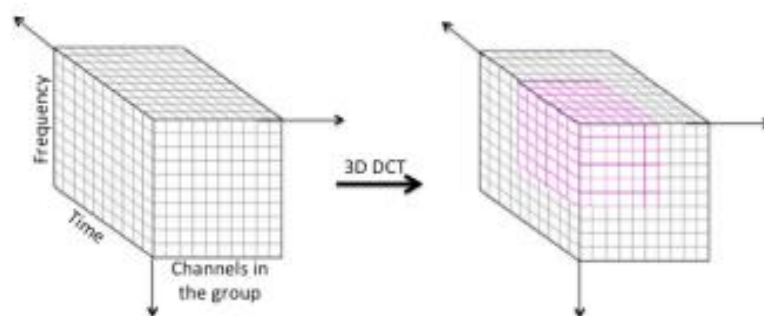


Figura 2: Rappresentazione grafica dell'operazione di reshaping



Altro fondamentale aspetto da precisare ai fini di una migliore comprensione della struttura del dataset dei biosegnali è la definizione stessa delle feature. Come discusso nel paragrafo precedente, ciascuna feature è in realtà costituita da diversi range di valori numerici non sempre raccolti in maniera omogenea. In particolare, a tutte le feature sono associate le seguenti misure statistiche:

- media;
- deviazione standard, che misura lo scostamento dei valori dal valor medio;
- asimmetria, che misura l'asimmetria della distribuzione dei valori rispetto al valor medio;
- curtosi, che misura l'appiattimento o l'allungamento della distribuzione dei valori rispetto alla distribuzione gaussiana (normale);
- percentuale dei valori  $> \text{media} + \text{deviazione standard}$ ;
- percentuale dei valori  $< \text{media} - \text{deviazione standard}$ .

## 2.2 Rappresentazione della Conoscenza

Definiti i biosegnali raccolti per il riconoscimento delle emozioni e le loro feature utili per capire come essi siano descritti attraverso il linguaggio matematico e statistico, è opportuno introdurre come sia possibile rendere queste stesse informazioni comprensibili per un computer.

### 2.2.1 Logiche Descrittive

Alla base del problema vi è il fatto che la conoscenza si definisce come informazione disponibile per la decisione e l'azione da parte di un sistema automatico. Per ottenerla, i linguaggi standard per rendere le informazioni *machine readable* (leggibili da calcolatore) sono insufficienti, occorre che l'informazione sia *machine understandable* (interpretabile da calcolatore).

Prendendo come esempio il linguaggio XML (Extensible Markup Language)<sup>3</sup>, ampiamente utilizzato per la rappresentazione di dati, è possibile codificare il valore di un singolo dato in maniera molto generale utilizzando delle etichette di testo con la sintassi seguente:

```
<costo>85</costo>
```

Lo stesso dato, però, può essere codificato all'interno di un documento differente come:

```
<prezzo>85</prezzo>
```

---

<sup>3</sup> <https://www.w3.org/XML/>

Questo si verifica perché il linguaggio XML codifica e standardizza la *sintassi* dei dati (dunque la loro struttura), ma non la loro *semantica* (cioè il loro *significato*). Ciò significa che, anche se gli utenti di questi sistemi informativi sanno che ci si sta riferendo alla stessa informazione, un elaboratore non potrà comprendere che “prezzo” e “costo” sono sinonimi.

Da questo semplice esempio, risulta chiaro come sia necessario utilizzare strumenti che siano in grado di *rappresentare* la conoscenza, permettendo ai computer di comprendere il significato dei dati stessi. La Rappresentazione della Conoscenza è una branca dell'Intelligenza Artificiale fornisce metodi di rappresentazione delle informazioni in maniera tale che siano machine understandable. Questi metodi si basano principalmente su due componenti fondamentali, ovvero un *linguaggio per la rappresentazione di conoscenza* e un sistema di *procedure di ragionamento*. Il linguaggio di rappresentazione della conoscenza consiste in un linguaggio formale, con una sintassi testuale o grafica, le cui espressioni sono utilizzate per rappresentare informazioni di uno scenario di riferimento (*dominio*).

A prescindere dal tipo di rappresentazione adoperato (simbolica o dichiarativa, come la *logica dei predicati del primo ordine* – *First Order Logic, FOL*), e dalla sintassi e notazione che il linguaggio utilizza (da notazioni testuali nei libri di logica, fino a codifiche ASCII e/o sintassi XML nei dispositivi di calcolo), è importante precisare come occorra che le espressioni utilizzate abbiano un significato definito in modo univoco.

Tra i linguaggi attualmente più utilizzati per la Rappresentazione della Conoscenza vi è la famiglia delle Logiche Descrittive (DL) [19]. Si tratta di *linguaggi terminologici*, cioè linguaggi atti a definire vocabolari strutturati e dotati di semantica formale, detti *terminologie* od *ontologie*. Elementi fondamentali di una DL sono:

- *classi* (o *concetti* o *termini*), che rappresentano insiemi di elementi nel frammento di realtà che si intende descrivere, detto *universo* del discorso;
- *proprietà* (o *ruoli*), che denotano relazioni tra coppie di elementi;
- *individui* (o *nominali*), cioè specifiche istanze di classi.

Questi elementi possono essere combinati per formare espressioni di concetti. Ciascuna DL è caratterizzata da un certo numero di operatori e costruttori logici scelti in un repertorio di operatori possibili. L'espressività di una determinata ontologia sarà legata all'insieme di costruttori e operatori che si sceglie di adoperare. Un'ontologia è costituita da un insieme di assiomi terminologici di equivalenza o sussunzione tra concetti. Una ontologia (anche detta *Terminological Box*, TBox) insieme ad un set di individui che ad essa si riferiscono (*Assertion Box*, ABox), costituisce una *Base di Conoscenza*

(*Knowledge Base*, KB). Per le espressioni di una DL è possibile specificare una semantica formale, che associa a ogni termine e a ogni enunciato terminologico un'interpretazione definita in modo insiemistico.

Un esempio di logica descrittiva è la logica  $\mathcal{ALN}$  (*Attributive Language with Number Restrictions*), utilizzata in questo lavoro di tesi. Essa utilizza i seguenti costrutti per definire i concetti:

- *Congiunzione di concetti* ( $C \sqsubseteq A \sqcap B$ ): indica che il concetto  $C$  è contemporaneamente sottoclasse di  $A$  e  $B$ ;
- *Quantificatore universale qualificato* ( $\forall R.C$ ): indica che se un individuo partecipa alla relazione  $R$  l'altro individuo che partecipa alla relazione deve essere istanza del concetto  $C$ .
- *Restrizioni numeriche non qualificate* ( $\leq n R$ ) ( $\geq n R$ ): indicano rispettivamente il massimo e il minimo numero di volte che le istanze del concetto devono partecipare alla relazione  $R$ . Sono usate solitamente (ma non obbligatoriamente) in coppia con il corrispettivo quantificatore universale sulla stessa relazione.

L'utilità di  $\mathcal{ALN}$  risiede nel fatto che, nonostante non permetta di utilizzare alcuni costrutti previsti in logiche con maggior espressività, permette comunque di descrivere la realtà di interesse attraverso un linguaggio leggero e flessibile, che costituisce un ottimo compromesso tra l'espressività e la complessità di elaborazione.

Tra i linguaggi utilizzati per la rappresentazione della conoscenza è possibile citare OWL (Web Ontology Language), standard proposto dal World Wide Web Consortium (W3C<sup>4</sup>) per la definizione di ontologie per il Web semantico.

### 2.2.2 Servizi di Inferenza

Come si è visto, una base di conoscenza permette di rappresentare delle conoscenze comprensibili dall'utente umano in modo tale che siano anche interpretabili per un elaboratore. Ciò che contraddistingue una KB, rispetto ad una base di dati è la possibilità di eseguire delle *procedure di ragionamento* in modo automatico sulla conoscenza in essa contenuta.

---

<sup>4</sup> W3C home page: <https://www.w3.org>

Il termine “ragionamento” in questo caso si riferisce ad algoritmi di tipo deduttivo, dove la *deduzione* (o *inferenza*) consiste nel verificare se un enunciato  $X$  è una conseguenza logica di una base di conoscenza. È importante distinguere tre concetti fondamentali per comprendere in cosa consiste il ragionamento su basi di conoscenza:

1. *compito di ragionamento*: è caratterizzato dal tipo di enunciati che si desidera dedurre da una base di conoscenze;
2. *procedura di ragionamento*: è l'algoritmo che consente la deduzione;
3. *servizio di ragionamento*: un servizio effettivamente implementato da uno strumento e messo a disposizione dalle applicazioni che accedono alla base di conoscenza.

Ad esempio, sia l'ABox che la TBox possono essere oggetto di specifici compiti di ragionamento. I principali compiti sull'ABox definiti in letteratura – e perciò chiamati *standard* – sono:

- *instance check*: dati una TBox  $\mathcal{T}$ , un'ABox  $\mathcal{A}$ , un termine arbitrario  $C$  e un nominale  $a$ , stabilire se  $\mathcal{T}, \mathcal{A} \models C(a)$ , cioè se da  $\mathcal{T}$  ed  $\mathcal{A}$  deriva come conseguenza logica che  $a$  è istanza di  $C$ ;
- *retrieval*: dati una TBox  $\mathcal{T}$ , un'ABox  $\mathcal{A}$  e un termine arbitrario  $C$ , fra tutti i nominali presenti nella base di conoscenza trovare tutti i nominali  $a_1, \dots, a_n$  tali che  $\forall k=1, \dots, n: \mathcal{T}, \mathcal{A} \models C(a_k)$ ;
- *realizzazione*: dati una TBox  $\mathcal{T}$ , un'ABox  $\mathcal{A}$ , un insieme di termini arbitrari  $\{C_1, \dots, C_n\}$  e un nominale  $a$ , determinare gli  $m$  termini  $\{C_{i1}, \dots, C_{im}\}$  più specifici in  $\{C_1, \dots, C_n\}$  per cui si ha che  $\mathcal{T}, \mathcal{A} \models C_k(a)$ .

I compiti di ragionamento standard relativi alla TBox, invece, sono i seguenti:

- *sussunzione*: data una TBox  $\mathcal{T}$  e due concetti  $C$  e  $D$  in  $\mathcal{T}$ , stabilire se la sussunzione  $C \sqsubseteq D$  è conseguenza logica di  $\mathcal{T}$ , in formule  $\mathcal{T} \models C \sqsubseteq D$ ;
- *equivalenza*: data una TBox  $\mathcal{T}$  e due concetti  $C$  e  $D$  in  $\mathcal{T}$ , stabilire se l'equivalenza  $C \equiv D$  è conseguenza logica di  $\mathcal{T}$ , ovvero stabilire se  $\mathcal{T} \models C \equiv D$ ;
- *soddisfacibilità*: data una TBox  $\mathcal{T}$ , stabilire se un termine  $C$  è soddisfacibile, nel senso che l'insieme degli individui dell'universo che soddisfano il termine non è necessariamente vuoto;
- *disgiunzione*: data una TBox  $\mathcal{T}$ , stabilire se due termini  $C$  e  $D$  sono disgiunti, nel senso che l'intersezione degli insiemi di individui dell'universo che soddisfano i due termini è necessariamente vuota.

Per quanto concerne le *procedure di ragionamento*, spesso si fa ricorso a teoremi di equivalenza per ridurre la soluzione di un compito a quella di un altro: ad esempio è possibile ridurre tutti i compiti di ragionamento sulla TBox alla sola soddisfacibilità o alla sola sussunzione. In scenari complessi come il riconoscimento delle emozioni, tuttavia, i compiti di ragionamento standard sulla TBox forniscono solo risposte di tipo “sì/no” e pertanto sono in grado di individuare unicamente match esatti oppure nessun match: ciò risultano inadeguato in contesti in cui occorre confrontare espressioni articolate come quelle che descrivono le feature dei biosegnali estratti da un soggetto e i modelli di classificazione delle emozioni. Sfruttando ulteriori compiti di ragionamento cosiddetti *non-standard* il matchmaking semantico è in grado di misurare il grado di (mancata) corrispondenza tra due espressioni di concetti in DL, che assumono rispettivamente il ruolo di *richiesta* e *risorsa* in uno scenario di ricerca (*discovery*), e di stabilire un ranking di un insieme di risorse sulla base della distanza semantica (e quindi della rilevanza) rispetto ad una data richiesta.

I più ampiamente utilizzati ragionatori per il Semantic Web presenti in letteratura e nell'industria sono progettati principalmente per operazioni *batch* su grandi ontologie e linguaggi molto espressivi, con ampie risorse di calcolo: ciò li rende non adatti ad utilizzi più flessibili ed in contesti pervasivi.

Per superare tali limitazioni, è possibile adottare servizi di inferenza *non-standard* su motori di ragionamento appositamente progettati per sistemi di calcolo mobili ed embedded. In particolare il lavoro di tesi usa i servizi forniti da *Mini-ME (Mini Matchmaking Engine)* [2], un matchmaker semantico specificatamente sviluppato per supportare le tecnologie del Web Semantico in contesti pervasivi e ontologie basate sulla Logica Descrittiva  $\mathcal{ALN}$ .

Tali servizi [20], brevemente richiamati di seguito, permettono sia di ottenere una spiegazione a partire dalle operazioni di sussunzione e soddisfacibilità, e al contempo permette un ranking su base logica delle risorse rilevate durante l'operazione di ragionamento:

- **Concept Contraction:** data una richiesta  $R$  e una risorsa  $S$ , non compatibili tra loro, viene determinata quale parte di  $R$  è in conflitto con  $S$ . Viene dunque restituita una versione contratta  $K$  (per *Keep*) della richiesta, tale che  $K$  e  $S$  siano soddisfacibili, ed una espressione  $G$  (per *Give up*) che rappresenta la parte di  $R$  eliminata dalla contrazione, spiegando il motivo dell'incompatibilità tra  $R$  e  $S$ .

- **Concept Abduction:** se una richiesta  $R$  e una risorsa  $S$  sono compatibili, ma  $S$  non sussume  $R$ , l'adduzione permette di determinare cosa deve essere ipotizzato in  $S$  affinché  $R$  sia sussunta. Viene restituita una soluzione  $H$  (per Hypotesis) che identifica le cause della mancata sussunzione tra  $R$  e  $S$ , interpretabile come ciò che deve essere ipotizzato in  $S$  tale da sussumere  $R$ .

## 2.3 Ontologia per il riconoscimento delle emozioni

Una volta descritte le basi teoriche su cui si struttura il framework di riconoscimento delle emozioni oggetto del lavoro di tesi, è possibile illustrare ora l'idea complessiva del framework proposto per il riconoscimento su base semantica delle emozioni dal punto di vista pratico, dedicando particolare attenzione a quali sono stati gli step che hanno preceduto il lavoro di sviluppo del framework stesso.

L'obiettivo del lavoro è quello di costruire un sistema capace di riconoscere le emozioni in maniera totalmente automatizzata, sfruttando la conoscenza a priori contenuta in una base di conoscenza: a tal fine il dataset di riferimento sarà elaborato per annotare i puri dati numerici con espressioni di concetti della TBox di una ontologia in linguaggio ALN opportunamente modellata. Così è possibile generare una ABox che mappa il dataset di riferimento ed affrontare il problema del riconoscimento delle emozioni come un problema di matchmaking semantico, deducendo nuova conoscenza attraverso i servizi di ragionamento non-standard illustrati nel paragrafo precedente applicati alla base di conoscenza stessa.

Per quanto riguarda la base di conoscenza, è molto importante visualizzare e comprendere la struttura del TBox, contenente la conoscenza concettuale da cui è partito il lavoro. La TBox della KB presenta la struttura di alto livello illustrata in Figura 3:

- Una radice **Thing**, che in linguaggio OWL rappresenta il concetto più generico, da cui tutte le classi (del tipo *owl:Class*) che si possono rappresentare devono derivare, per definire il dominio dell'ontologia;
- Una classe **Biosignal**, a partire da cui sono modellati tutti i biosegnali inclusi ed utilizzati nel dataset di riferimento (MEG, ECG, hEOG, T\_EMG) (Figura 4);
- Una classe **Emotion**, che rappresenta la radice per tutte le emozioni che si intende riconoscere attraverso il framework, così come definite dallo studio DECAF: *Amusing*, *Angry*, *Disgusting*, *Happy*, *Exciting*, *Fear*, *Funny*, *Sad*, *Shock* (Figura 5);
- Una classe **Feature** da cui discendono tutte le singole feature relative ad ogni biosegnale. Questa sezione della tassonomia risulta essere la più articolata (Figura 6).

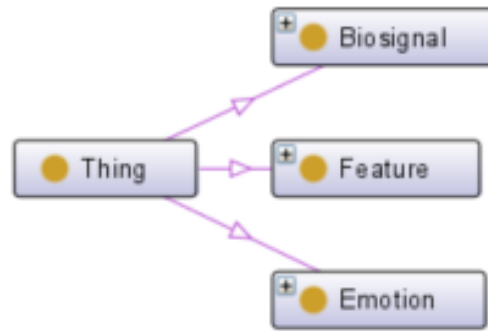


Figura 3: Classi di primo livello presenti dell'ontologia

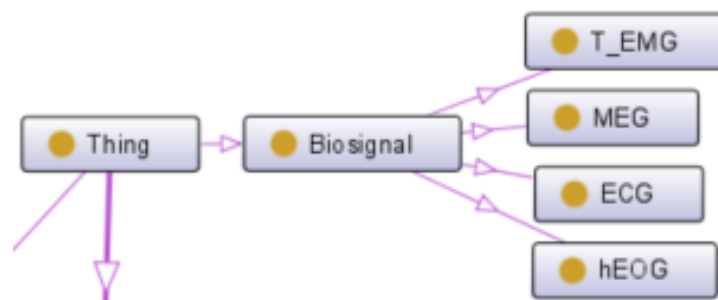


Figura 4: Modellazione della classe Biosignal

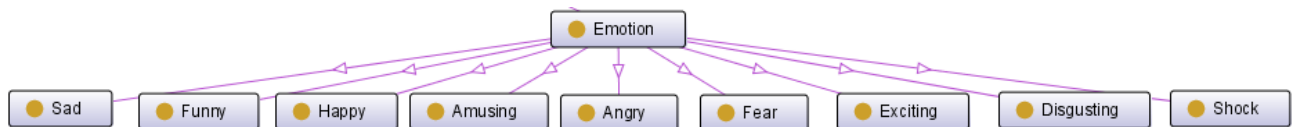


Figura 5: Estratto dell'ontologia con la classe Emotion espansa

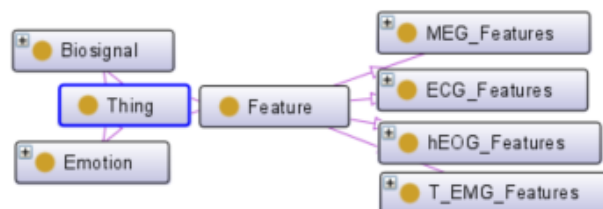


Figura 6: Estratto dell'ontologia con la classe Feature espansa

Per ottenere la formalizzazione delle feature sui quattro tipi di biosegnali analizzati, in primo luogo è stato condotto un lavoro di ordinamento di dati e clustering [22], a partire dalle matrici descritte nel

paragrafo 2.1.3, contenenti tutti i biosegnali raccolti e le feature calcolate per ciascun esperimento relativo ad ogni paziente nel dataset DECAF.

Il lavoro si è svolto attraverso due fasi:

1. nella prima sono stati realizzati dei vettori contenenti per ogni video, per ognuno degli indici (statistici e analitici) considerati per ogni feature, i biosegnali estratti da tutti i pazienti;
2. nella seconda è stato effettuato il clustering dei dati per ciascun indice statistico di ogni feature, ottenendo il numero di cluster ottimale con il metodo del gomito, il valore del centroide di ogni cluster e il valore minimo e massimo di ogni cluster.

Mentre l'algoritmo di produzione dei vettori per le prime tre matrici di biosegnali è molto simile, è stato prestato un trattamento diverso per il biosegnale relativo alla Magnetoencefalografia, essendo composto da due vettori di celle separati, uno relativo ai dati dei magnetometri e l'altro relativo a quelli dei gradiometri.

Nel lavoro condotto sul dataset preliminarmente alla tesi, l'algoritmo di clustering utilizzato è stato il *K-Means*, il quale partiziona un dataset in *K* cluster distinti e non sovrapposti. In particolare, per ogni indice statistico, l'ultima fase di clustering è stata effettuata in due maniere differenti:

1. considerando tutti i video che ogni paziente ha visionato;
2. utilizzando il raggruppamento secondo l'emozione suscitata come descritto in [8].

Oltre alla produzione degli algoritmi necessari per svolgere le operazioni di *data cleaning* e analisi statistica dei dati, durante questa fase preliminare è stata effettuata anche la produzione di una matrice aggiuntiva, contenente i valori anomali rilevati nei risultati. In diversi casi, infatti, l'analisi ha evidenziato una forte rumorosità dei dati, con valori mancanti, valori fisicamente impossibili e outlier. Si è cercato di ridurre l'elevato numero di anomalie presenti, al fine di riuscire ad avere un dataset il più omogeneo possibile, attraverso l'analisi statistica dei valori e l'analisi grafica. Durante l'analisi dei valori anomali, infatti, ci si è accorti che spesso per ogni singolo esame era solo un gruppo ristretto di soggetti a generare la maggior parte delle anomalie. Si è dunque giunti all'ipotesi che tali soggetti possano aver provocato la maggior parte di questi valori anomali per tre motivi principali:

- errato posizionamento degli elettrodi;
- stato sovraeccitato del soggetto;
- errata calibrazione della strumentazione.

Per migliorare la qualità dei dati si è quindi ricorsi a due metodi:



1. analisi statistica, attraverso la quale il numero delle anomalie si è ridotto di oltre il 70%;
2. analisi grafica, ricorrendo all'utilizzo degli outliers, identificando quali persone hanno provocato più outliers stabilendo una soglia empiricamente oltre la quale il soggetto venisse escluso. Tale metodo ha ridotto dell'80% il numero di valori indesiderati.

Rimossi questi valori anomali, sono stati sostituiti con la media dei restanti valori, in modo tale da mantenere invariata la struttura delle matrici e non alterare significativamente la distribuzione dei dati raccolti. Il dataset ripulito risultante è stato nuovamente sottoposto a clustering.

L'operazione di individuazione dei valori anomali, raggruppamento e clustering è stata dunque il punto di partenza del successivo lavoro di generazione della base di conoscenza. La seconda fase della produzione dell'ontologia si è poi incentrata sull'aggiunta dei cluster ottenuti per ogni feature di ogni biosegnale e sui relativi valori di minimo e massimo.

Nel popolamento dell'ontologia, ad ogni cluster del dataset pulito è stata fatta corrispondere una sottoclasse della classe foglia che ne rappresenta la feature. Il popolamento ha incluso sia tutte le emozioni, sia tutte le feature dei biosegnali considerati nella sperimentazione, con il proprio range di valori possibili.

Ciascuna emozione è descritta da una combinazione di range di feature, rilevati durante la visione della clip. Ogni range verrà poi tradotto nella classe corrispondente, che nell'ontologia di dominio ne modella il significato. Per quanto riguarda invece il popolamento delle feature dei biosegnali, per ciascun biosegnale è stata rispettata la struttura e la nomenclatura del dataset DECAF descritta nel paragrafo 2.1.3. Per ottenere il risultato in Figura 7, la descrizione dell'ontologia di dominio è stata realizzata utilizzando il linguaggio OWL 2 con sintassi OWL/XML. Ogni classe inserita nell'ontologia, per rispettare la struttura stabilita, ha dovuto essere:

1. **dichiarata** utilizzando il seguente elemento OWL/XML:

```
<Declaration>
  <Class IRI="#NomeClasse"/>
</Declaration>
```

2. **Definita** come sottoclasse del concetto padre che identifica la feature, nel seguente modo:

```
<SubClassOf>
  <Class IRI="#NomeClasse"/>
  <Class IRI="#ClassePadre"/>
</SubClassOf>
```

3. **Annotata** inserendo per ogni classe due *annotation property* che definiscono il valore minimo e massimo registrato nel cluster, nel seguente modo:

```
<AnnotationAssertion>
  <AnnotationProperty IRI="#maxValue"/>
  <IRI>#NomeClasse</IRI>
  <Literal datatypeIRI="http://www.w3.org/2001/
XMLSchema#float">valoreMax</Literal>
</AnnotationAssertion>

<AnnotationAssertion>
  <AnnotationProperty IRI="#minValue"/>
  <IRI>#NomeClasse</IRI>
  <Literal datatypeIRI="http://www.w3.org/2001/
XMLSchema#float">valoreMin</Literal>
</AnnotationAssertion>
```

dove *NomeClasse* è l'IRI (*Internationalized Resource Identifier*) della specifica sottoclasse che si vuole generare. Un esempio di annotation property degli estremi di un cluster è illustrato in Figura 8 nell'editor di KB *Protégé* [23] e in Figura 9 come testo in sintassi OWL/XML.

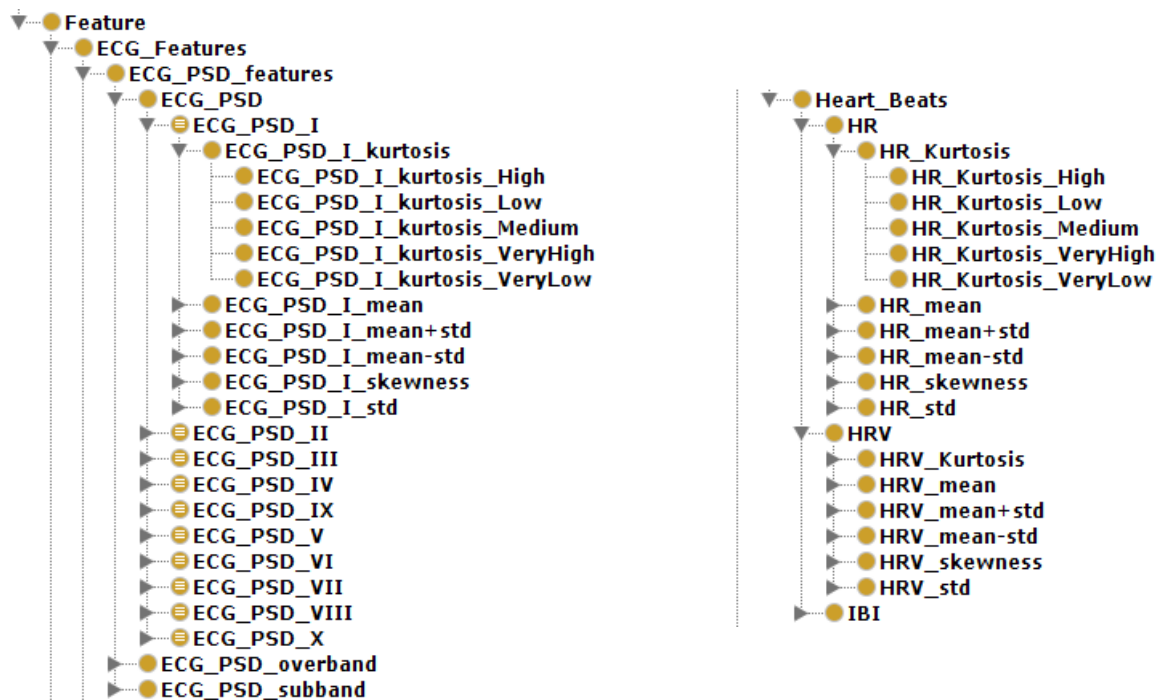


Figura 7: Estratto di ECG features con visualizzazione in Protégé

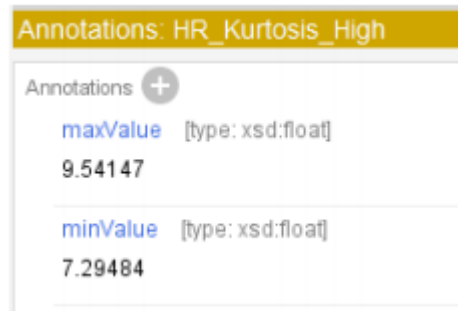


Figura 8: Intervallo di valori [min,max] relativo a una singola feature

```
<AnnotationAssertion>
  <AnnotationProperty IRI="#maxValue"/>
  <IRI>#HR_Kurtosis_High</IRI>
  <Literal
datatypeIRI="http://www.w3.org/2001/XMLSchema#float">9.54147
</Literal>
</AnnotationAssertion>

<AnnotationAssertion>
  <AnnotationProperty IRI="#minValue"/>
  <IRI>#HR_Kurtosis_High</IRI>
  <Literal
datatypeIRI="http://www.w3.org/2001/XMLSchema#float">7.29484
</Literal>
</AnnotationAssertion>
```

Figura 9: Intervallo di valori [min,max] relativo a una singola feature, in formato OWL/XML

Ogni *NomeClasse* rappresenta un singolo cluster e il pattern di assegnazione dei nomi dipende dal numero di cluster ottenuti attraverso lo step di lavoro precedente, relativo alla generazione dei cluster di ogni feature. In particolare, i suffissi da aggiungere all'IRI dipendono direttamente dal numero di cluster ottenuti dal metodo del gomito:

- 2 cluster: *Low, High*
- 3 cluster: *Low, Medium, High*
- 4 cluster: *Low, MediumLow, MediumHigh, High*
- 5 cluster: *Low, MediumLow, Medium, MediumHigh, High*
- 6 cluster: *VeryLow, Low, MediumLow, MediumHigh, High, VeryHigh*
- 7 cluster: *VeryLow, Low, MediumLow, Medium, MediumHigh, High, VeryHigh*
- 8 cluster: *ExtremelyLow, VeryLow, Low, MediumLow, MediumHigh, High, VeryHigh, ExtremelyHigh*

- 9 cluster: *ExtremelyLow*, *VeryLow*, *Low*, *MediumLow*, *Medium*, *MediumHigh*, *High*, *VeryHigh*, *ExtremelyHigh*

Da Figura 10 a Figura 13 è mostrato un esempio di annotazione delle feature per ognuno dei quattro biosegnali analizzati in DECAF.

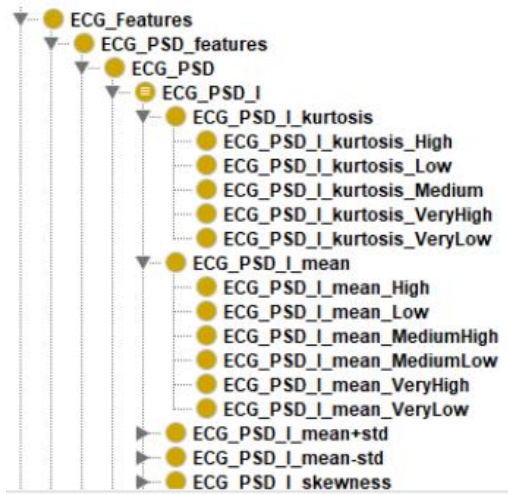


Figura 10: Esempio di feature ECG

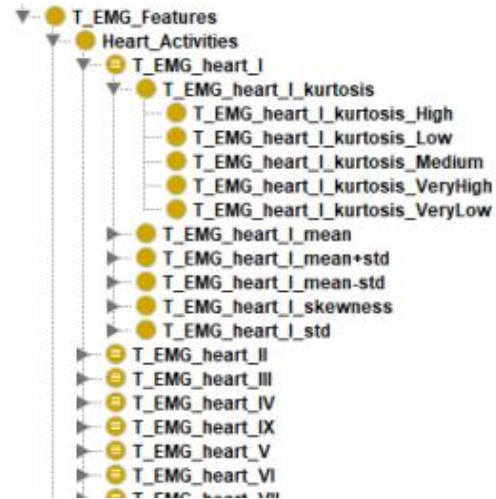


Figura 11- Esempio di feature EMG

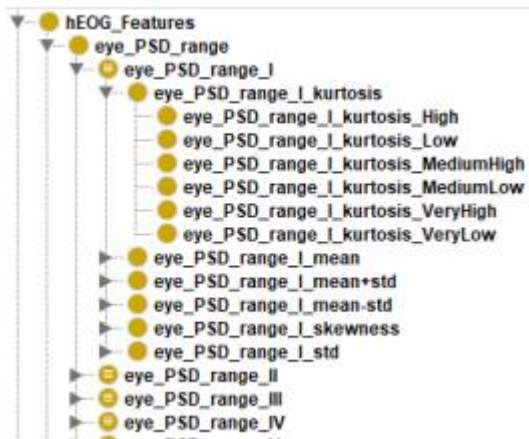


Figura 12: Esempio di feature hEOG

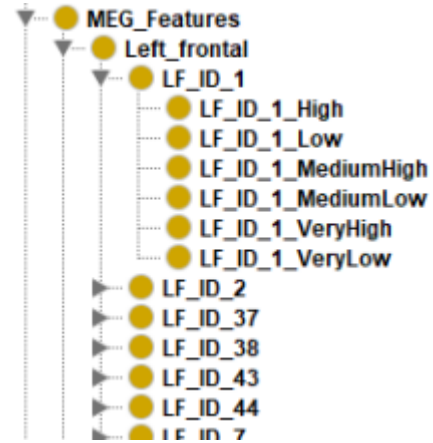


Figura 13: Esempio di feature MEG

## Capitolo 3

### Riconoscimento delle emozioni mediante inferenza automatizzata

#### 3.1 Descrizione del problema

Il capitolo espone il lavoro compiuto in questo lavoro di tesi per la progettazione e l'implementazione del framework di riconoscimento delle emozioni attraverso ragionamento automatico. Dal punto di vista pratico, tale lavoro ha richiesto diversi step e l'utilizzo di più strumenti software.

Per effettuare il popolamento dell'ABox a partire dalla formalizzazione della TBox descritta nel capitolo precedente sono stati rilevati ed analizzati tutti i valori di feature di biosegnali utilizzando il software MATLAB, con cui è avvenuta anche l'annotazione e la nomenclatura delle emozioni rilevate dagli esperimenti come congiunzione di queste feature, utilizzando la sintassi OWL/XML.

A questo scopo è stato necessario estrarre dall'ontologia gli intervalli numerici [minVal, maxVal] relativi alle varie classi delle feature generate in maniera automatica come illustrato nel paragrafo 2.3.

Concluso il popolamento dell'ABox, sono state definite le seguenti operazioni per:

1. Effettuare il Matchmaking semantico utilizzando come richiesta una quota parte di emozioni di *training*, e come risorsa una quota parte di emozioni di *test*.
2. Valutare attraverso uno studio statistico i risultati del matchmaking e l'utilizzo di metodi di rappresentazione leggibili e sintetici.

#### 3.2 Generazione dell'ABox

La prima parte del lavoro si è incentrata sulla generazione dell'ABox a partire dai dati contenuti nel dataset DECAF andando a formalizzare all'interno della base di conoscenze ogni singola esperienza di emozione come congiunzione di tutte le feature relative ai biosegnali considerati. Per fare ciò, l'elaborazione dei dati necessari attraverso MATLAB è stata suddivisa in due fasi:

1. generazione di un *training set* e di un *test set* che dividessero in modalità *hold out* l'intero dataset per il riconoscimento automatizzato delle emozioni;

2. popolamento dell'ABox attraverso l'annotazione delle emozioni, rispettando la sintassi OWL/XML di OWL 2.

Per quanto riguarda la prima fase, il lavoro di estrapolazione dei dati si è basato sulle matrici fornite dai precedenti studi effettuati sul dataset DECAF e descritte nel paragrafo 2.1.3.

Oltre al caricamento delle strutture dati fondamentali, è stato creato un ambiente di elaborazione contenente tutte le strutture aggiuntive per facilitare il successivo lavoro di confronto dei dati ed annotazione.

In primo luogo è stata creata una lista di emozioni che rispettasse quella contenuta nella documentazione del dataset, secondo la corrispondenza tra emozioni e ID dei clip video mostrata in Figura 14.

```
emotion_clips = {
    'Amusing', [1 2 4 5];
    'Funny', [3 6 7 9];
    'Happy', [8 10 11 12 13 14 16 17 18];
    'Exciting', [15 33];
    'Angry', [19 21 23 35];
    'Disgusting', [28 34];
    'Fear', [30 36];
    'Sad', [20 22 24 25 26 27 31];
    'Shock', [29 32]; };
```

Figura 14: Lista degli ID di clip relativi a ciascuna emozione

Sfruttando la funzione *cvpartition*<sup>5</sup> built-in di MATLAB, è stato possibile facilitare il lavoro di divisione del dataset in *training set* e *test set*. La funzione permette infatti di dividere, utilizzando il parametro *HoldOut*, una qualsiasi struttura dati in una parte di test e una di training, definendo la porzione di test che si desidera avere. Nella fattispecie sono state stabilite le proporzioni di 30% del dataset come test set ed il restante 70% come training set. È stato deciso di applicare la funzione direttamente alla *PermutationList* di Figura 14 per ottenerne due partizioni, conservando comunque la casualità dell'emozione considerata nei singoli esperimenti di DECAF. Inoltre, si è voluto garantire la casualità delle istanze di test e di training utilizzando la funzione built-in *rng*<sup>6</sup>, che ha permesso di definire un seed casuale per l'estrazione delle singole entità. Tale divisione è stata necessaria poiché l'obiettivo del lavoro era quello di sviluppare un framework in grado di riconoscere autonomamente le emozioni attraverso i servizi di inferenza, e dunque sfruttando le emozioni contenute nel training set per costruire un modello

<sup>5</sup> <https://it.mathworks.com/help/stats/cvpartition.html>

<sup>6</sup> <https://it.mathworks.com/help/matlab/ref/rng.html>

– precisamente, un’ABox – da cui fosse possibile riconoscere attraverso il matchmaking semantico le emozioni nelle istanze del test set somministrate al sistema.

Le emozioni di test sono pertanto tutte quelle di cui i soggetti casuali scelti all’interno della PermutationList avranno fatto esperienza, per ogni clip video visionata. Questi soggetti *test* saranno identificati da un indice “1” ottenuto da un vettore di valori binari ottenuto come risultato della funzione *cvpartition*, fondamentale per la fase successiva di popolamento.

Dopo aver effettuato la partizione, la prima fase di generazione ha avuto luogo con l’iterazione su PermutationList di ogni singola visione di clip da parte di ogni paziente. L’emozione relativa alla clip indice dell’iterazione è stata facilmente reperibile confrontando se l’ID di clip ottenuto rientrasse in una delle emozioni nella lista precedentemente indicata. In questo modo è stato possibile identificare l’emozione che si stava analizzando all’interno dell’iterazione sia da un punto di vista numerico che con una rappresentazione testuale, necessaria per la fase successiva.

Fissate dunque la *clip* e la *emotion* all’interno della procedura, segue la logica di estrazione effettiva dei dati. Per semplificare questa operazione sono state create una o più matrici contenenti solo ed unicamente tutti i valori di feature di biosegnali relativi a quella singola esperienza di emozione, situata in ciascuna delle matrici tridimensionali originarie dei biosegnali. Ricordando che ciascuna delle matrici è caratterizzata da una struttura  $NS \times NV \times NFeature$ , estraendo a priori il soggetto (riga di PermutationList) e la clip (colonna di PermutationList e casuale per ogni soggetto), è stato possibile creare quattro matrici di feature di biosegnali specifiche per ogni emozione. È stato necessario porre particolare attenzione alla parte di estrazione dei dati relativi ai risultati di Magnetoencefalografia, raccolti nella complessa struttura descritta nel paragrafo 2.1.3. Come anche affermato nel capitolo precedente, è stato infatti necessario effettuare un *reshaping* della matrice dei biosegnali per estrarre i  $2 \times 2 \times 2$  coefficienti di interesse, come mostrato nel codice in Figura 15.

```
Tmp1 = reshape(1:216,6,6,6);
FeatureIDs = reshape(Tmp1(1:2,1:2,1:2),8,1);
TheFirst8Features = SLDCT_GRADfeatures{SubjectID}( FeatureIDs, :, : );
```

Figura 15: Reshaping della matrice  $6 \times 6 \times 6$  di MEG

Riadattando questa operazione ed integrandola nel processo di reshaping, si ottiene il codice riportato in Figura 16.

```
ecg_values(:) = ECGFeatures(subject,clip,:);
emg_values(:) = EMGFeatures(subject,clip,:);
eog_values(:) = EOGFeatures(subject,clip,:);
Tmp1 = reshape(1:216,6,6,6);
```

```
FeaturesIDs = reshape(Tmp1(1:2,1:2,1:2),8,1);
meg_values = SLDCT_GRADfeatures{subject}([1 2 7 8 37 38 43 44],clip,:);
```

Figura 16: Vettorizzazione delle matrici per una singola emozione

Si può notare, però, che questa operazione ha prodotto una valorizzazione complessiva di ogni biosegnale. Prendendo ad esempio *ecg\_values*, questo sarà un vettore  $1 \times 92$ , con 92 valori di feature relativi al biosegnale specifico per l'emozione che si sta analizzando. La stessa osservazione può essere condotta sugli altri biosegnali. È dunque necessario effettuare un ulteriore reshape di ogni singolo vettore per ottenere i sotto-vettori di interesse, strutturati in modo conforme alla documentazione DECAF. Mentre per i valori MEG è stato possibile sotto-vettorizzare i valori in maniera più semplice, poiché suddivisi in 9 intervalli omogenei di 8 feature, per ECG è stato necessario suddividere in maniera più esplicita ciascun sotto-vettore, poiché non possono godere di una struttura omogenea. Per EMG ed ECG è stato possibile utilizzare la stessa funzione di *reshape* adoperata per MEG.

Per ultimare la prima fase di estrazione ed ordinamento dei dati, sono state infine create 4 liste relative ai 4 biosegnali, come nell'esempio di Figura 17, per rendere più facile il lavoro di popolamento e confronto dei valori di feature.

```
hEOG_Features = {
    "eye_PSD_range_I", eog_features(1,:);
    "eye_PSD_range_II", eog_features(2,:);
    "eye_PSD_range_III", eog_features(3,:);
    "eye_PSD_range_IV", eog_features(4,:);
    "eye_PSD_range_V", eog_features(5,:);
    "eye_PSD_range_VI", eog_features(6,:);
    "eye_PSD_range_VII", eog_features(7,:);
    "eye_PSD_range_VIII", eog_features(8,:);
    "zig_PSD_range_I", eog_features(9,:);
    "zig_PSD_range_II", eog_features(10,:);
    "zig_PSD_range_III", eog_features(11,:);
};
```

Figura 17: Un esempio di lista di feature di biosegnale hEOG, costituita da un elenco di sotto-vettori del vettore *eog\_values*

La seconda fase del lavoro per la creazione dell'ABox è costituita dal popolamento effettivo della ABox. Anch'essa è stata svolta utilizzando MATLAB.

Innanzitutto, sono stati definiti i pattern che la formalizzazione della conoscenza deve seguire affinché possa essere correttamente aggiunta all'ABox in riferimento all'ontologia descritta in 2.3. Ogni



emozione, che sia di test o di training, diventerà una espressione di concetti all'interno dell'ontologia, seguendo l'elemento base riportato in Figura 18.

```

<ClassAssertion>
  <ObjectIntersectionOf>
    <ObjectSomeValuesFrom>
      <ObjectProperty IRI="#ObjectProperty_nameN"/>
      <Class abbreviatedIRI="owl:Thing"/>
    </ObjectSomeValuesFrom>
    <ObjectAllValuesFrom>
      <ObjectProperty IRI="#ObjectProperty_nameN"/>
      <Class IRI="#Feature_name_n"/>
    </ObjectAllValuesFrom>
  </ObjectIntersectionOf>
  <NamedIndividual IRI="#NameEmotionN"/>
</ClassAssertion>

```

Figura 18: Modellazione di un individuo in sintassi OWL/XML all'interno dell'ontologia

In particolare:

- **ObjectProperty\_nameN** indica il nome della proprietà relativa al biosegnale di una singola istanza di emozione (es. ECG\_Features). Saranno dunque 4 per ogni istanza di emozione;
- **Feature\_name\_n** indica il nome specifico della feature appartenente al biosegnale, rispettando sia la struttura stabilita da DECAF, sia la nomenclatura utilizzata nel precedente lavoro di popolamento. Questi elementi possono variare da 0 a  $n$  in base alla quantità di feature individuate per il biosegnale di riferimento.

Dal punto di vista implementativo, per la lettura e la scrittura sul file relativo all'ontologia è stata utilizzata la funzione *xmlread*<sup>7</sup>, attraverso cui è stato possibile trattare la base di conoscenze come un semplice file XML, implementando ogni classe dell'ontologia con le varie funzioni offerte da MATLAB. Ogni chiave è stata aggiunta attraverso la funzione *ontology.createElement('key')*, dove *ontology* è il file della KB letto attraverso MATLAB e *key* è il particolare nome di attributo in formato XML (nel nostro caso può essere "ClassAssertion", "ObjectIntersectionOf", ecc.).

Per quanto riguarda la formalizzazione delle singole feature, la loro nomenclatura nell'ontologia è del tipo:

<sup>7</sup> <https://it.mathworks.com/help/matlab/ref/xmlread.html>

`<Segnale>_<Range>_<Feature>_<Cluster>`

dove *Segnale* è il nome del biosegnale, *Range* indica il range di appartenenza della feature nel segnale (I, II, III...), *Feature* indica il singolo parametro statistico di feature (skewness, stddev, kurtosis, etc.) e *Cluster* indica il tipo di valore nel cluster di appartenenza. Mentre è stato possibile identificare con un piccolo numero di iterazioni i primi 3 suffissi utilizzando le liste precedentemente definite, per identificare il suffisso di *Cluster* è stato necessario confrontare il valore della feature all'interno di tutti i possibili intervalli di valori accettabili. In particolare, in maniera iterativa è stato confrontato ogni valore di feature estratto con ogni valore minimo e massimo relativo ad ogni cluster possibile, al fine di identificare il cluster specifico.

```
for feature in Features
  for cluster in ClustersOfFeatures
    if feature not NaN and cluster not NaN
      if feature >= cluster(min_value) and feature <= cluster(max_value)
        class = ontology.createClass;
        class.setIRI(feature_name + cluster_name);
        ontology.append(class);
```

Figura 19: Pseudo-codice per l'identificazione della feature in un cluster specifico

Nella porzione di pseudo-codice in Figura 19, relativa unicamente alle feature del biosegnale MEG, viene identificata la feature esatta verificando che il valore si trovi in almeno uno dei cluster disponibili. Per quelle feature il cui numero di cluster possibili è inferiore a 9, eventuali intervalli [minValue, maxValue] saranno stati riportati come NaN, e dunque sarà necessario verificare che l'intervallo che si controlla non sia di quel tipo. Da notare, inoltre, le ultime righe di codice in cui viene definito l'IRI completo della feature, come stringa composta *FeatureName+SuffissoCluster*.

Completando le operazioni di scrittura sul file, e specificando se l'emozione in questione era di test o di training, è stato possibile popolare in maniera completamente automatizzata l'intera ABox dell'ontologia a partire dal dataset DECAF.

Per identificare ogni istanza di emozione, è stata utilizzata la seguente nomenclatura:

- **Training Set:** #<emozione>\_<IDSoggetto>\_<IDclip>
- **Test Set:** #Test<emozione>\_<IDSoggetto>\_<IDclip>

Il risultato del processo di annotazione è mostrato in Figura 20 per un individuo del training set nella KB. È possibile notare la notevole estensione delle annotazioni e nel contempo la regolarità della loro struttura.

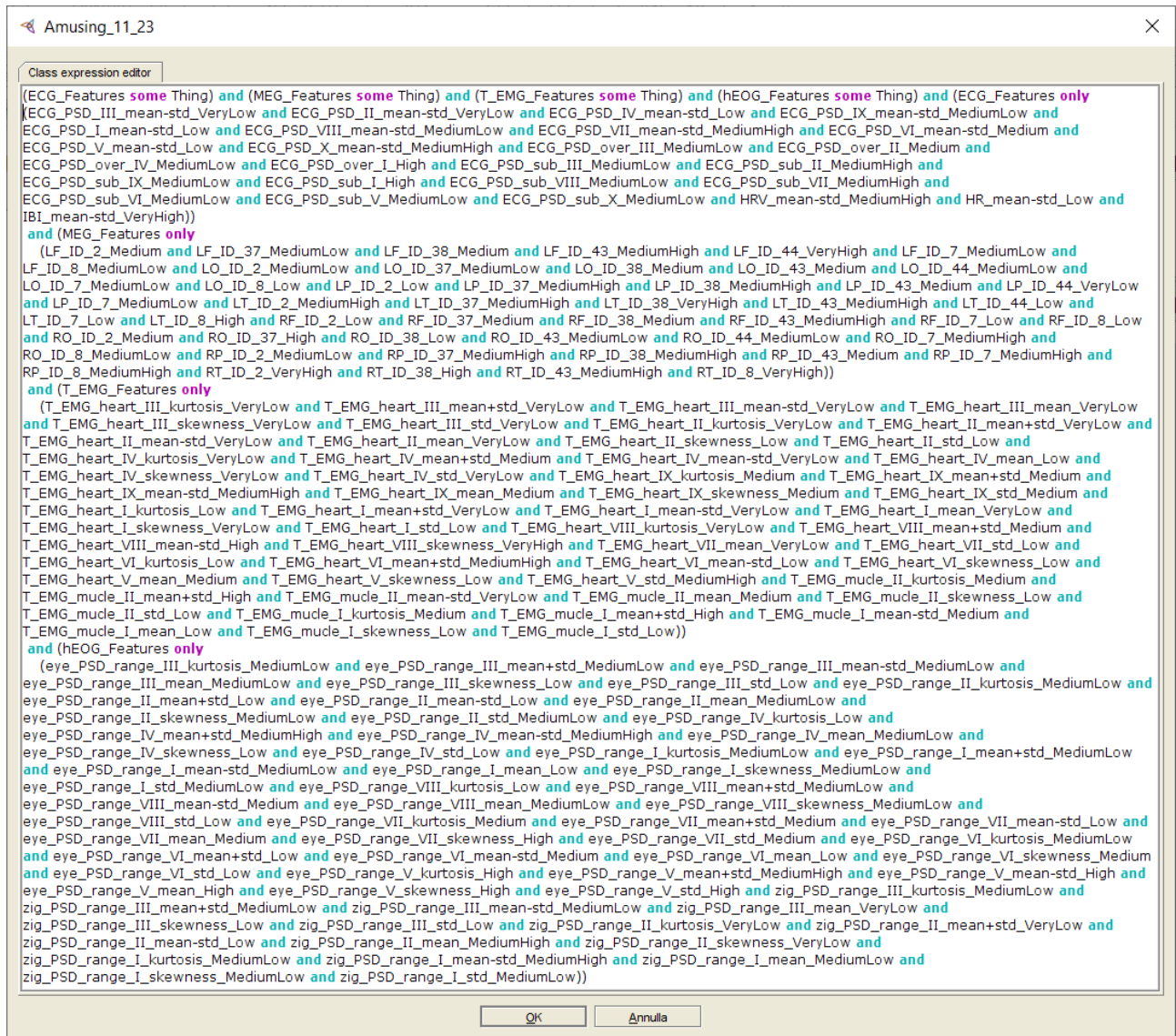


Figura 20: Esempio di individuo di training nell'ABox

### 3.3 Matchmaking semantico per il riconoscimento delle emozioni

Dopo aver generato la base di conoscenza completa, la seconda parte della progettazione del framework si è incentrata sullo sviluppo del sistema di matchmaking semantico automatizzato. Il linguaggio di programmazione scelto è stato Java, attraverso cui è stato possibile utilizzare i servizi di inferenza non-standard forniti da Mini-ME [2] [20]. In particolare, sono stati utilizzati i servizi di Concept Abduction

e Concept Contraction per riconoscere automaticamente le emozioni relative al Test Set precedentemente generato, attraverso un numero ben determinato di emozioni appartenenti al Training Set.

Nel caso specifico, dunque, le *risorse* nel problema di matchmaking sono costituite dalle emozioni di Test, poiché sono ciò che il sistema deve essere in grado di riconoscere a partire da emozioni di Training, che costituiranno le *richieste*. Utilizzando sia la Concept Contraction che la Concept Abduction è possibile ricavare una *penalty*, un valore numerico associato alla distanza dell'approssimazione risultante dal matchmaking semantico. Le emozioni sono ordinate in base al valore di *penalty* e quelle con valore più basso sono classificate come semanticamente più vicine e pertanto più verosimili nel problema di identificazione dell'emozione. Per rendere più robusta la classificazione dell'emozione, si è deciso di non prendere unicamente l'emozione relativa all'istanza semanticamente più vicina all'istanza di test, ma di utilizzare un approccio basato su *k-NN* (K-Nearest Neighbors) [23]. Tale approccio, comune nell'ambito del machine learning, consiste in un algoritmo di classificazione di oggetti sulla base della loro vicinanza rispetto ad uno considerato. In particolare, definito un oggetto di cui interessa ottenere una classificazione, si identifica la sua classe attraverso la maggioranza degli oggetti nel suo intorno *K*.

In questo modo, riconoscere attraverso matchmaking semantico un'emozione significa ottenere l'emozione richiesta “più vicina”, in termini di *penalty* minima, che si presenta con frequenza maggiore tra i risultati del *k-NN*.

Dal punto di vista implementativo, sono state create delle variabili di ambiente destinate a facilitare il processo di computazione:

- un intero *k*, parametro dell'algoritmo, corrispondente al numero di individui che si vogliono utilizzare nel *k-NN*: sono stati scelti tre valori per *k*=1, *k*=5, *k*=10;
- una lista contenente le emozioni note;
- 3 header differenti utilizzati all'interno di 3 file in formato CSV (Comma Separated Values), prodotti per mostrare i risultati del matchmaking corrispondenti alle tre configurazioni relative ai 3 valori di *K*.

Per ogni confronto effettuato, il sistema memorizza le informazioni schematizzate in Tabella 5 e descritte di seguito:

TestIndividual	ToE(ms)	MajorityIndividual	miPenalty	ClosestIndividual_n	ciPenalty_n
----------------	---------	--------------------	-----------	---------------------	-------------

Tabella 5: Elenco delle informazioni memorizzate nel file, per colonna

1. **TestIndividual**: rappresenta il nome dell'emozione risorsa;

2. **Time of Execution (ToE)**: calcolata in millisecondi tra l'inizio e la fine dell'operazione di Abduction per una emozione di test per ogni emozione di training;
3. **MajorityIndividual**: rappresenta l'individuo di maggioranza, ovvero la cui frequenza è massima nell'configurazione kNN;
4. **miPenalty**: indica la penalty dell'individuo di maggioranza;
5. **ClosestIndividual\_n**: rappresenta l'emozione richiesta più vicina semanticamente in termini di penalty (cioè con penalty minore);
6. **ciPenalty\_n**: rappresenta la penalty relativa al ClosestIndividual.

Per l'configurazione  $k=1$  MajorityIndividual coinciderà con ClosestIndividual, mentre nei casi  $k=\{5,10\}$  si avranno rispettivamente 5 ClosestIndividual e 10 ClosestIndividual in ordine crescente di penalty.

Le operazioni necessarie per la produzione dei file in formato .CSV contenenti i risultati delle Abduction, per i valori di  $k=\{1, 5, 10\}$ , hanno seguito quest'ordine:

1. generazione del ragionatore Mini-ME, come istanza della classe *ReasonerFactory*, per il caricamento della KB (in formato .owl);
2. scomposizione dell'ABox in individui di tipo Resource (corrispondenti alle emozioni da riconoscere, dunque contenenti la voce "Test" nel loro IRI) e Request (corrispondenti alle emozioni utilizzate come modello per il riconoscimento);
3. verifica della compatibilità tra la Resource e tutte le Request, mediante iterazione, con conservazione dei risultati delle Abduction in una lista;
4. ordinamento di ciascuna lista dei risultati in base al valore di penalty ottenuto, in ordine crescente, per le Abduction relative ad una data Resource;
5. calcolo del tempo di esecuzione totale (ToE) per il matchmaking semantico relativo alla singola Resource
6. determinazione dell'entità Request più vicina (*ClosestIndividual*) e dell'emozione classificata a maggioranza (*MajorityIndividual*) per ogni individuo di test.

Compilando il codice per valori di  $k=\{1,5,10\}$  sono prodotti automaticamente i file CSV, il cui nome è "log\_k-NN.csv" dove NN è il valore di k scelto. In Tabella 6 e Tabella 7 si riportano due estratti di log, rispettivamente per k minimo e k massimo.

Un'analisi approfondita dei risultati ottenuti sarà discussa nel capitolo successivo, alla luce di test sperimentali sistematici.

TestIndividual	ToE(ms)	ClosestIndividual_1	ciPenalty_1
TestFunny_21_1	85	Shock_30_30	107.0
TestExciting_22_36	39	Exciting_11_30	80.0
TestFunny_21_6	27	Shock_16_15	86.0
TestFear_19_16	27	Shock_30_30	72.0
TestSad_21_16	26	Shock_6_33	81.0
TestSad_21_19	28	Shock_6_33	97.0

Tabella 6: Estratto di log per  $k = 1$ 

TestIndividual	ToE(ms)	MajorityIndividual	ciMajorityIndividual	ClosestIndividual_1	ciPenalty_1	ClosestIndividual_2	ciPenalty_2
TestFunny_21_1	57	Shock_30_30	107.0	Shock_30_30	107.0	Shock_6_33	107.0
TestExciting_22_36	40	Exciting_11_30	80.0	Exciting_11_30	80.0	Exciting_30_1	93.0
TestFunny_21_6	30	Happy_16_4	100.0	Shock_16_15	86.0	Shock_6_33	95.0
TestFear_19_16	29	Shock_30_30	72.0	Shock_30_30	72.0	Happy_30_10	83.0
TestSad_21_16	28	Shock_6_33	81.0	Shock_6_33	81.0	Happy_16_4	85.0

Tabella 7: Estratto di log per  $k=10$ , fino all'individuo più vicino di ordine 2

## Capitolo 4

### Esperimenti

#### 4.1 Metodologia sperimentale

Per ottenere una rappresentazione più chiara dei dati raccolti ed effettuare analisi statistiche per valutare la bontà dei risultati, sono state effettuati differenti test sperimentali per ogni configurazione scelta.

Le fasi relative allo studio statistico, fissato un valore di  $k$  per ogni  $k$ -NN, sono state le seguenti:

1. Ordinamento di tutti gli individui di test per emozione (ottenendo dunque la classe “vera”);
2. Costruzione della matrice di confusione [22] relativa alle  $T \times P$  emozioni ottenute attraverso il matchmaking semantico (dove  $T$  coincide con le classi “vere” e  $P$  coincide con le classi “predette” per maggioranza);
3. Calcolo, per ogni classe, degli indici di prestazione dell'algoritmo: *Precision*, *Recall*, *Accuracy*, *F1-score*.

Nella matrice di confusione su ogni riga sono state poste le *true classes*, ovverosia le emozioni “vere” che si intendeva riconoscere attraverso il matchmaking semantico, mentre sulle colonne sono state poste le *predicted classes* (classi predette), che rappresentano per ciascuna emozione di test l’emozione classificata risultata dal matchmaking.

In particolare, in una matrice di confusione ogni cella rappresenta la quantità (o la frequenza, se in forma percentuale) di istanze predette per ciascuna classe rispetto alla classe vera da riconoscere.

L’obiettivo della matrice di confusione è quello di mostrare, in una maniera sintetica ma molto efficace, la bontà dei risultati ottenuti attraverso operazioni di classificazione. In Tabella 8 si riporta un esempio di matrice di confusione relativa alla configurazione=10.

Oltre alla semplice visualizzazione sintetica matriciale, e alla quantizzazione in percentuale dei risultati, la matrice di confusione ci ha anche permesso di approfondire i risultati dei test attraverso dei parametri di valutazione. Per specificarli è innanzitutto necessario dare le seguenti definizioni generali relative ad una matrice di confusione per una classificazione binaria:

- per **true positive** si intende la corretta identificazione della classe predetta (corrispondente, dunque, alla diagonale principale);

- per **true negative** si intendono le predizioni correttamente rifiutate per una determinata classe (corrispondenti a tutti gli individui predetti in tutta la matrice escludendo quelli contenuti nella colonna e nella riga di una classe specifica);
- per **false positive** si intendono le predizioni incorrettamente effettuate relativamente ad una classe specifica (corrispondenti a tutti gli elementi della colonna relativi ad una classe, escluso quello sulla diagonale principale);
- per **false negative** si intendono le predizioni incorrettamente rifiutate per una determinata classe (corrispondenti agli individui appartenenti alla riga di una stessa classe, escluso quello sulla diagonale principale).

TC \ PC	Amusing	Angry	Disgusting	Exciting	Fear	Funny	Happy	Sad	Shock
Amusing	13.89%	0%	0%	0%	0%	0%	58.33%	0%	27.78%
Angry	0%	19.44%	0%	0%	0%	0%	19.44%	0%	61.11%
Disgusting	0%	0%	16.67%	0%	0%	0%	27.78%	0%	55.56%
Exciting	0%	0%	0%	11.11%	0%	0%	44.44%	0%	44.44%
Fear	0%	0%	0%	0%	11.11%	0%	5.56%	0%	83.33%
Funny	0%	0%	0%	0%	0%	11.11%	44.44%	0%	44.44%
Happy	0%	0%	0%	0%	0%	0%	74.07%	1.23%	24.69%
Sad	0%	0%	0%	0%	0%	0%	20.63%	19.05%	60.32%
Shock	0%	0%	0%	0%	0%	0%	33.33%	0%	66.67%

Tabella 8: Matrice di confusione per  $k=10$ 

Alla luce di queste definizioni, sono stati calcolati i parametri di valutazione:

- **Accuracy:** totale di predizioni corrette diviso l'intero test set. In formule:  $\frac{TP + TN}{TP + TN + FP + FN}$
- **Precision:** il numero di individui identificati come positivi diviso il numero totale degli attuali positivi. In formule:  $\frac{TP}{TP + FP}$
- **Recall:** rapporto degli individui identificati correttamente come positivi diviso il numero totale di individui identificati come positivi. In formule:  $\frac{TP}{TP + FN}$
- **F1-score:** variazione armonica di Precision e Recall. È calcolata come  $\frac{2 \times Precision \times Recall}{Precision + Recall}$  e rappresenta l'efficacia della classificazione.

Una volta calcolati tutti gli FN, FP, TP e TN necessari, sono stati calcolati i parametri e rappresentati in forma tabellare per ogni emozione. Un esempio è osservabile in Tabella 9.



	Accuracy	Precision	Recall	F1-Score
<b>Amusing</b>	0,904	1	0,139	0,244
<b>Angry</b>	0,91	1	0,194	0,326
<b>Disgusting</b>	0,954	1	0,167	0,286
<b>Exciting</b>	0,951	1	0,111	0,2
<b>Fear</b>	0,951	1	0,111	0,2
<b>Funny</b>	0,901	1	0,111	0,2
<b>Happy</b>	0,698	0,438	0,741	0,55
<b>Sad</b>	0,84	0,923	0,19	0,316
<b>Shock</b>	0,552	0,079	0,667	0,142

Tabella 9: Indici di prestazione per  $k = 10$

Tutti i test sono stati eseguiti su un computer desktop con le seguenti specifiche:

- Sistema Operativo Windows 10 Pro 64-bit;
- CPU Intel Core i5-4460 con 4 core e frequenza di clock 3.20 GHz;
- RAM 8,00 GB dual-channel DDR3 a 799 MHz;
- Hard disk SATA 931 GB.

## 4.2 Risultati

Per quanto riguarda l'efficienza della computazione, a partire dai log contenenti l'elenco di tutti i problemi di matchmaking semantico risolti, sono stati ricavati i valori di Time of Evaluation per ogni singola Abduction. Raccogliendo tali valori per ogni classificazione sono stati calcolati il tempo di esecuzione medio, minimo e massimo, riportati di seguito in Tabella 10.

È evidente che i tempi di esecuzioni medi sono estremamente bassi, considerando che il matchmaking per ogni individuo di test ha compreso il confronto con 756 individui del training set. Questo rappresenta un dato più che positivo a livello di elaborazione dei dati.

	Mean ToE (ms)	Min ToE (ms)	Max ToE (ms)
<b>k = 1</b>	27,343653	19	85
<b>k = 5</b>	29,185759	21	91
<b>k = 10</b>	29,256966	20	101

Tabella 10: Tempo di esecuzione del matchmaking semantico in millisecondi

Attraverso l'utilizzo delle matrici di confusione è stato invece possibile approfondire, oltre al semplice risultato del matchmaking, anche le prestazioni dell'intera fase di classificazione in termini di efficacia.

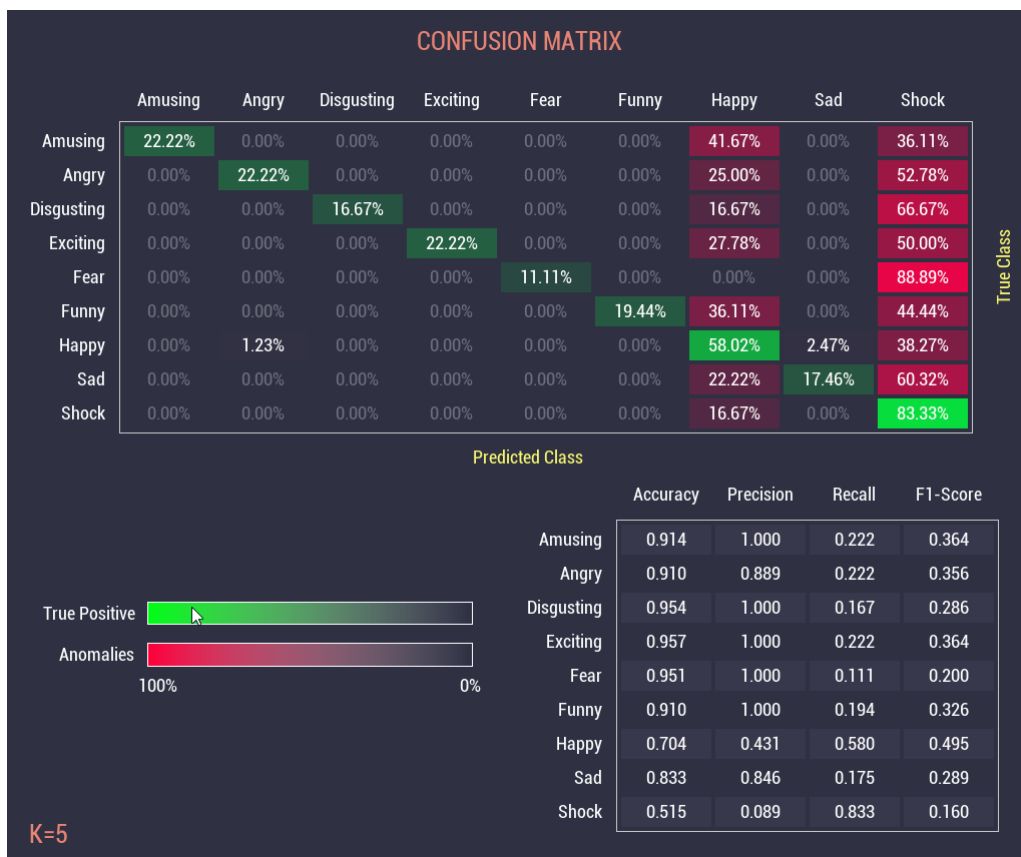
A livello intuitivo è possibile comprendere che un risultato migliore di matrice di confusione dovrebbe tendere ad avere valori maggiori sulla diagonale principale della matrice, poiché sulla diagonale si trovano le corrispondenze *predicted class = true class*. Le matrici di confusione ottenute mettono però in evidenza alcuni errori sistematici di classificazione, all'interno dei quali la maggior parte delle emozioni non viene correttamente riconosciute. La presenza di tali anomalie è evidenziata dai risultati ottenuti in Figura 21, Figura 22 e Figura 23, rilevati rispettivamente per  $k=1$ ,  $k=5$  e  $k=10$ .

Iniziando l'analisi dai risultati presentati nelle matrici di confusione, è possibile osservare che per le emozioni *Amusing*, *Angry*, *Disgusting*, *Exciting*, *Fear*, *Funny* e *Sad* la quantità di istanze correttamente predette risulta molto bassa in tutti e tre i test. In particolare, per queste emozioni la coincidenza tra classe predetta e classe vera non supera mai il 25%, un dato molto basso.

L'assenza di alti valori di corrispondenze tra classi predette e classi vere, per queste emozioni, è associata a un numero elevato di falsi positivi risultanti per le emozioni *Happy* e *Shock*. È possibile notare come per queste due emozioni si abbiano valori percentuali molto elevati di falsi positivi, distribuiti in maniera abbastanza omogenea su tutte le emozioni. Inoltre, in tutti e tre i casi di test, solo in corrispondenza di queste emozioni si sono ottenuti i più alti valori di predizioni esatte. Per valori di  $k$  tra 1 e 5 le predizioni in corrispondenza di *Shock* sono le più alte in assoluto, quasi pari al 90%. Considerando l'intervallo massimo di  $k=10$  invece, il numero di predizioni esatte relative a *Shock* si abbassa, ma in compenso aumentano le predizioni esatte di *Happy*.

Dallo studio dei parametri statistici è inoltre possibile osservare come, in corrispondenza dei dati anomali, vi siano i più bassi valori di Accuracy e Precision, in contrapposizione a valori molto alti di Recall ed F1-Score. Seppur piccolo, è comunque degno di nota anche il valore percentuale che assume la predizione dell'emozione *Sad* sull'emozione vera *Happy*, sempre presente in tutti i casi di test tra 1-2%.

Occorre ricordare che le matrici di confusione sono il risultato della classificazione eseguita con matchmaking semantico e basata sul valore di penalty che ogni Abduction produceva durante il confronto emozione per emozione. È dunque presumibile che la causa della presenza di valori percentuali molto alti in corrispondenza solo delle emozioni *Shock* ed *Happy* possa derivare da una loro preponderanza in termini di quantità all'interno dei risultati del matchmaking semantico.

Figura 21: Matrice di confusione e indici di prestazione per  $k=1$ Figura 22: Matrice di confusione e indici di prestazione per  $k=5$

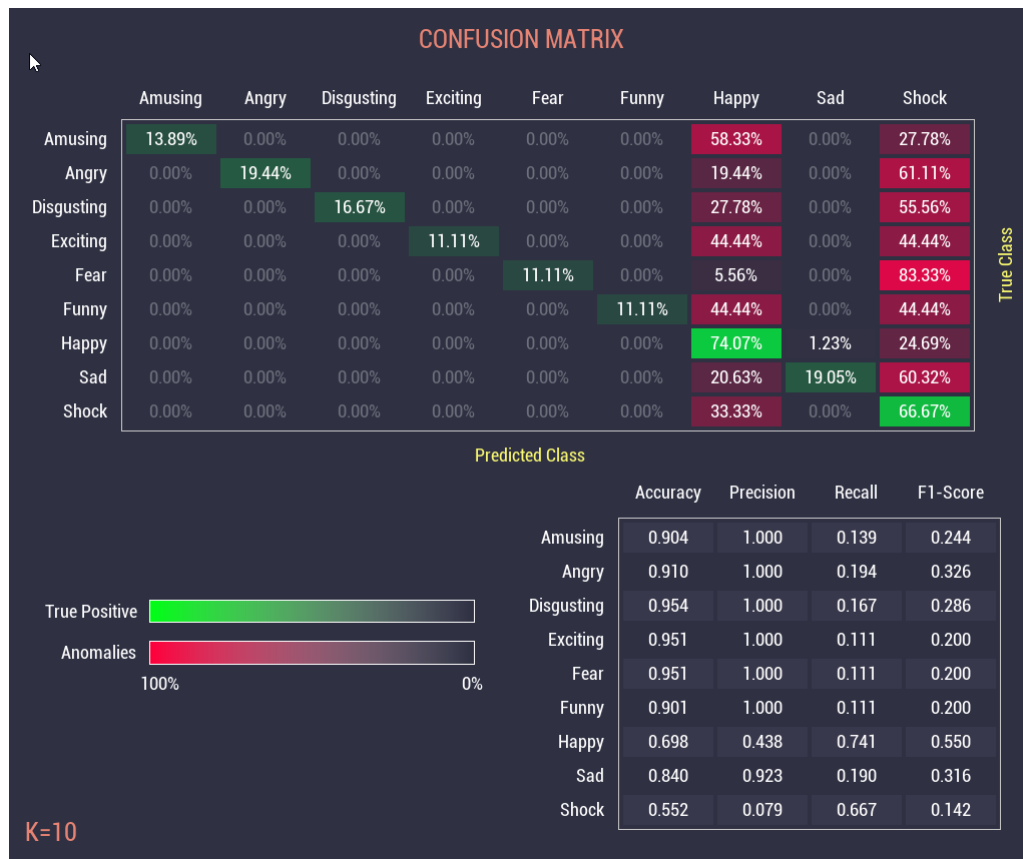


Figura 23: Matrice di confusione e indici di prestazione per  $k=10$

Per approfondire l'analisi dei risultati, sono stati esaminati i log della classificazione, da cui è possibile notare come la maggioranza dei risultati del matchmaking contenga valori di Majority Individual e Closest Individual non coincidenti all'emozione di Test. Se la presenza di un Majority Individual differente potrebbe, talvolta, consistere in un errore marginale ed accettabile (poiché rappresenta semplicemente l'emozione di training con penalty minore la cui frequenza nell'intervallo di  $k$  è maggiore), meno spiegabili sono tutti i Closest Individual (che rappresentano l'individuo in assoluto con penalty minore) errati. Si è tuttavia notato che un ristretto numero di individui del training set è risultato contenere l'individuo rilevato come più vicino dal matchmaking semantico in una frazione elevata di campioni di test:  $\{Schock\_6\_33, Schock\_30\_30, Happy\_30\_10\}$ . Ciò rappresenta certamente un risultato da valutare attentamente, poiché indica che nel matchmaking semantico queste istanze contengono annotazioni dotate di un significato molto vicino ad emozioni differenti.

Per avere una visione complessiva e quantificata di questi errori di classificazione, sono state eseguite le seguenti operazioni:

1. per ogni emozione di test, sono stati identificati i casi in cui l'emozione classificata risultasse essere errata;
2. sono stati raccolti e quantificati solo i casi in cui l'emozione classificata rilevata dal primo step coincidesse anche con l'emozione più vicina (mettendosi dunque nelle condizioni di poter considerare “errori accettabili” le emozioni di maggioranza non coincidenti né con l'emozione di test né con l'emozione più vicina);
3. sono stati raccolti e quantificati tutti questi casi, per singola emozione, cercando di identificare l'entità dell'anomalia.

Tali operazioni sono state effettuate per tutti i valori di  $k$ , con i risultati riportati in Tabella 11. Gli errori di classificazione (denotati come *PossibleAnomalies*) si presentano in quantità molto elevate. In particolare, per  $k=1$  (dove l'emozione più vicina e l'emozione classificata coincidono) le anomalie superano la metà dei test totali, mentre per i restanti valori di  $k$  si abbassano a meno della metà, pur restando in numero molto elevato. Per tutti le configurazioni, inoltre, risulta evidente la presenza di risultati costanti, come l'istanza del training set *Shock\_30\_30* la cui presenza è di 155/342 in tutti e tre i casi di studio; analogamente l'emozione *Shock\_6\_33* appare sempre 45/342 volte, ed anche *Happy\_6\_1* che, pur avendo la presenza al netto più bassa (solo 6 volte), rimane costante in tutti e tre i casi.

k = 1		k = 5		k = 10	
TotalTests	PossibleAnomalies	TotalTests	PossibleAnomalies	TotalTests	PossibleAnomalies
342	226	342	152	342	158
PossibleBugEntity	Count	PossibleBugEntity	Count	PossibleBugEntity	Count
Shock_30_30	155	Shock_30_30	155	Shock_30_30	155
Shock_16_16	52	Shock_16_16	52	Shock_16_33	45
Shock_16_33	45	Shock_16_33	45	Shock_16_15	28
Happy_30_10	28	Happy_30_10	28	Happy_30_10	28
Happy_16_4	29	Happy_16_4	29	Shock_18_5	60
Shock_18_5	60	Shock_18_5	60	Happy_6_1	6
Sad_11_35	2	Sad_11_35	2		
Happy_18_9	1	Happy_6_1	6		
Happy_6_1	6				

Tabella 11: Analisi e quantificazione delle anomalie per  $k = \{1, 5, 10\}$

L'origine della presenza eccessiva di tali anomalie sicuramente non può essere ricondotta ad una semplice maggioranza totale all'interno della KB. Ciò è evidente soprattutto combinando i dati delle matrici di confusione, i log e l'analisi delle anomalie: prendendo come esempio l'emozione *Amusing* nel caso  $k=10$ , solo il 13,89% delle emozioni predette corrispondono a tale emozione, mentre il 58,33% delle emozioni predette sono rappresentate da *Happy* ed il 27,78% da *Shock*. Ma riportando queste quantità in termini probabilistici, sapendo che nel dataset di training le emozioni *Amusing*, *Shock* ed *Happy* sono rispettivamente 83, 42 e 192, la relazione non sussiste.

Tali risultati sono dunque da imputare a delle caratteristiche semantiche specifiche di queste emozioni, tali da renderle *comuni*. Da un punto di vista semantico, se particolari feature sono sempre presenti significa che la loro descrizione nell'ontologia è tale da renderle molto generali e dunque assimilabili a feature di qualsiasi altra emozione. Alla radice vi è dunque un problema prettamente legato all'origine delle descrizioni delle emozioni, ovverosia come esse sono state generate in partenza.

Analizzando le emozioni raccolte, è stato notato come molte di queste presentino una descrizione abbastanza regolare e non molto differente dalla media. Inoltre, la maggior parte di queste emozioni anomale non presenta valori di feature legati al biosegnale MEG. Alcuni esempi sono mostrati di seguito da Figura 24 a Figura 27.

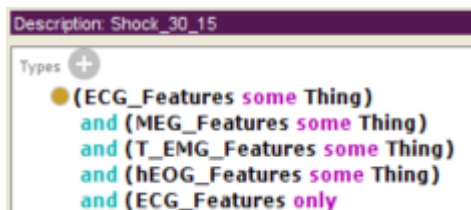


Figura 24: Emozione Shock\_30\_15, non anomala

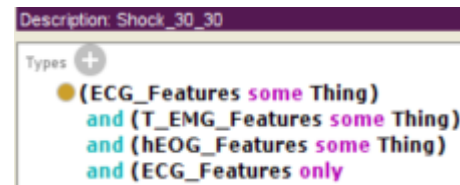


Figura 25: Emozione Shock\_30\_30, anomala

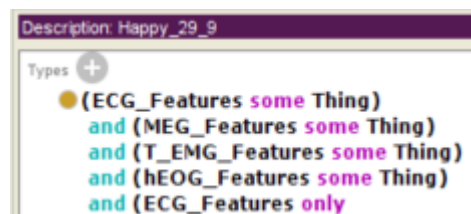


Figura 26: Emozione Happy\_29\_9, non anomala

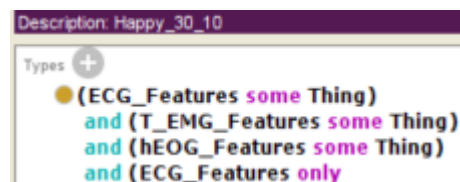


Figura 27: Emozione Happy\_30\_10, anomala

Questa seconda condizione è però dovuta all'assenza di valori nel dataset contenente le feature relative al biosegnale MEG. Infatti, attraverso una più approfondita analisi effettuata con MATLAB, verificando ogni singolo valore relativo ai gradienti contenuti nel vettore *SLDCT\_GRADfeatures*, è stata riscontrata

la presenza di valori NaN (*Not a Number*) in corrispondenza delle sotto-celle 2, 6, 16, 18, 22 e 30, su ogni riga delle colonne 19 e 23. È rilevante il fatto che la descrizione di alcune emozioni, proprio come *Schock\_30\_30*, la più presente in assoluto, risulti essere priva di valori di biosegnale MEG. Tale assenza rende le istanze di emozioni come *Schock\_30\_30* molto generiche e troppo simili a molte altre emozioni, non avendo dei valori specifici per uno dei quattro biosegnali di interesse.

Una volta identificata la possibile origine dei valori anomali, l'impossibilità di effettuare ulteriori operazioni di filtraggio e pulizia dei dataset ha condotto a tentare diverse strade per arginare il problema.

Una delle vie intraprese con l'obiettivo di migliorare i risultati ottenuti è stata consequenziale all'identificazione e alla quantizzazione delle anomalie presenti nel matchmaking: sono stati effettuati nuovi cicli di matchmaking, cosiddetti “nobugs”, all'interno dei quali sono stati esclusi alcuni degli individui anomali identificati. L'elenco degli individui eliminati è riportato in Figura 28.

Per fare ciò, è stata definita nel framework del matchmaking una lista di queste emozioni, in maniera tale che potessero essere condotti diversi test in maniera molto rapida ed elastica, escludendo qualsiasi servizio di inferenza che contenesse queste emozioni come richiesta. In tal modo si è condotto il processo di matchmaking semantico per ogni configurazione come se queste emozioni non fossero modellate nell'ontologia, al costo di ridurre il pool di individui disponibili per il riconoscimento automatizzato delle emozioni.

```
Static String[] exclude_bugs = {
    "Shock_30_30", "Shock_16_15", "Shock_18_5", "Happy_30_10", "Happy_16_4",
    "Shock_18_9", "Happy_6_1", "Sad_11_35", "Sad_30_3", "Sad_5_31", "Sad_11_15" };
```

*Figura 28: alcune anomalie escluse dopo 1: 2 cicli iterativi*

A seguito di questa operazione, sono state ricalcolate le nuove matrici di confusione riportate in Figura 29. Dopo il primo ciclo di identificazione e rimozione degli individui di training anomali, i risultati della classificazione sono essere migliorati. Anche se solo poche delle predicted classes superano il 50% per le corrispondenti true classes, sulla diagonale si hanno valori più alti rispetto al caso precedente.

Allo stesso tempo però, si può notare come la riduzione delle anomalie abbia portato uno *shifting* degli errori di classificazione da destra verso sinistra, con una distribuzione abbastanza omogenea. Avendo eliminato le anomalie preponderanti, infatti, risultano ora apparire tante anomalie più piccole, che probabilmente condividono la stessa origine con le precedenti, ma in scala ridotta.

Ciò si traduce in alcuni valori mancanti di feature, che rendono le descrizioni delle emozioni troppo generiche e facilmente confondibili.

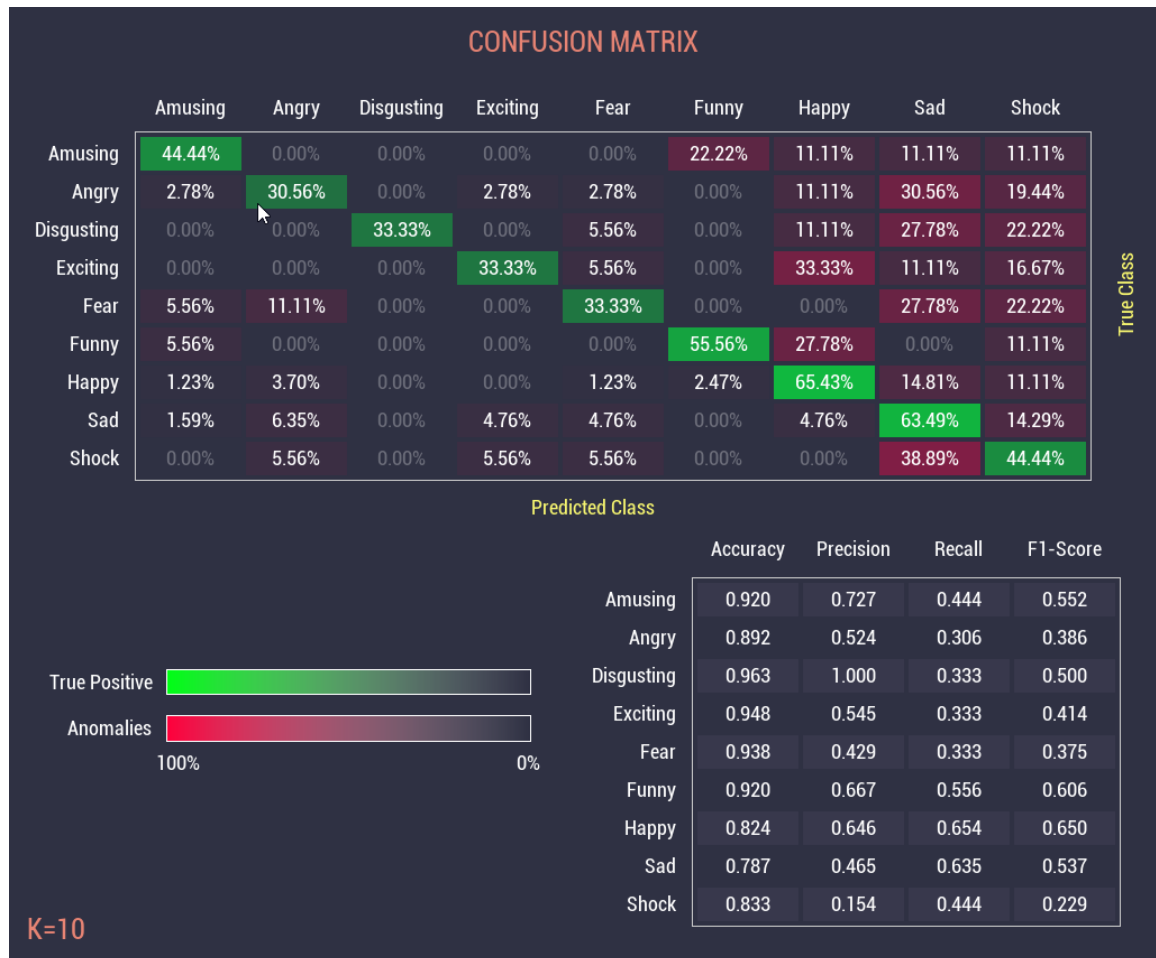


Figura 29: Matrice di confusione per  $k=10$  dopo l'esclusione di alcune anomalie



## Capitolo 5

### Conclusioni

Il lavoro di tesi ha avuto come obiettivo la realizzazione di un framework di riconoscimento automatizzato delle emozioni attraverso servizi di inferenza non-standard basati su tecniche di Rappresentazione della Conoscenza per la modellazione di biosegnali relativi a stimoli emotigeni.

Le attività realizzate nel lavoro di tesi includono:

- progettazione e implementazione di un framework MATLAB per la creazione automatizzata di una ABox riferita ad una ontologia in linguaggio OWL, a partire da un dataset di biosegnali puramente numerico e privo di semantica machine-understandable;
- classificazione di emozioni attraverso servizi di inferenza non-standard forniti da un motore inferenziale semantico (Mini-ME), che ha permesso di applicare operazioni di ragionamento deduttivo su di una base di conoscenza complessa e di dimensioni considerevoli in tempi ridotti;
- analisi dei risultati ed identificazione delle criticità del sistema, che ha permesso di identificare ed analizzare gli errori di classificazione ottenuti, consentendo anche di discutere riguardo alla risolvibilità dei problemi riscontrati.

Attraverso la sperimentazione condotta è stato possibile individuare un problema di classificazione, imputabile direttamente alla fase di modellazione del data set utilizzato come riferimento ed in particolare ai valori non presenti per alcune feature di alcuni individui. Queste evidenze rappresentano il punto di partenza per futuri miglioramenti del sistema a partire da un'attenta rivisitazione della fase di pre-processing, che crei un dataset più pulito per portare a dei risultati migliori.

A seguito di operazioni di filtraggio dei dati e raffinamento del dataset, lì dove i biosegnali presentano valori incerti o mancanti (che sia per problematiche fisiche verificatesi durante l'acquisizione, oppure per errori commessi durante l'annotazione dei valori stessi) non è stato comunque possibile ottenere risultati soddisfacenti negli esperimenti condotti.

Ulteriori sviluppi futuri del framework potrebbero includere:

1. implementazione di un filtro di riconoscimento dei valori anomali (identificabili con valore NaN) per l'approssimazione di questi valori a valori di feature medi (Medium High, Medium Low, ...) affinché presentino comunque valori di feature generalizzati ed accettabili.

2. implementazione di un meccanismo di analisi ed approssimazione dei valori anomali, attraverso un'osservazione statistica ed accurata di tutti gli altri valori di feature, creando così valori di feature “misti” basati su modelli statistici realistici per la stessa feature del medesimo biosegnale nell'ambito del dataset.

Entrambi gli approcci sopra accennati andrebbero studiati approfonditamente, implementati e valutati sperimentalmente per selezionare quello più vantaggioso.

## Bibliografia

- [1] E. Cambria. Affective Computing and Sentiment Analysis, *IEEE Intelligent Systems*, vol. 31, no. 2, pp. 102-107, 2016.
- [2] F. Scioscia, M. Ruta, G. Loseto, F. Gramegna, S. Ieva, A. Pinto, E Di Sciascio. Mini-ME matchmaker and reasoner for the Semantic Web of Things. In *Innovations, Developments, and Applications of Semantic Web and Information Systems*, pp. 262-294. IGI Global, 2018.
- [3] M. Ruta, F. Scioscia, A. Cinquepalmi, S. Cipriani, E. Di Sciascio. Dai biosegnali agli stati emotivi: un approccio semantico, XVI Convegno Nazionale Associazione Italiana Ingegneri Clinici (AIIC 2016), Apr 2016.
- [4] R. W. Picard. *Affective Computing*. MIT press. 2000
- [5] J. D. Mayer, P. Salovey. The intelligence of emotional intelligence. *Intelligence*, vol. 17, n. 4, 433-442, 1993.
- [6] D. Goleman. *Emotional Intelligence*, Bantam Books, 2006.
- [7] K. T. Johnson, R. W. Picard. Customizable, Interactive Toy Platform to Enable Motivation-Driven Cognitive and Physical Development in Children Diagnosed with Autism or Developmental Disorders. *ACM SIGCHI Interaction Design & Children*, pp. 149-158, 2017.
- [8] M. K. Abadi, R. Subramanian, S. M. Kia, P. Avesani, I. Patras, N. Sebe. DECAF: MEG-based multimodal database for decoding affective physiological responses, *IEEE Transactions on Affective Computing*, 6(3), 209-222, 2015.
- [9] P. Ekman. *Emotion in the human face*, Pergamon Press, 1982.
- [10] M. Bradley. Emotional memory: a dimensional analysis. In *Emotions: Essays on Emotion Theory*. Lawrence Erlbaum, Hillsdale, NJ, 1994.
- [11] M. Bradley, P. J. Lang. Measuring emotion: the self-assessment manikin and the semantic differential. *Journal of behavior therapy and experimental psychiatry*, 25(1), 49-59, 1994.
- [12] N. Watanabe, M. Yamamoto. Neural mechanisms of social dominance. *Frontiers in neuroscience*, vol. 9, article 154, 2015.
- [13] S. Koelstra, C. Muhl, M. Soleymani, J.-S. Lee, A. Yazdani, T. Ebrahimi, T. Pun, A. Nijholt, I. Patras. DEAP: A database for emotion analysis using physiological signals. *IEEE Transactions on Affective Computing*, 3(1):18–31, 2012.
- [14] N. Sebe, M. S. Lew, I. Cohen, Y. Sun, T. Gevers, T. S. Huang. Authentic Facial Expression Analysis. *Image and Vision Computing*, 25(12):1856–1863, 2007.

- [15] P.-Y. Oudeyer. The production and recognition of emotions in speech: features and algorithms. *International Journal of Human Computer Interaction*, 59(1-2):157–183, 2003.
- [16] U. Hasson, R. Malach, D. J. Heeger. Reliability of cortical activity during natural stimulation. *Trends in Cognitive Sciences*, 14(1):40-48, 2010.
- [17] M. Soleymani, J. Lichtenauer, T. Pun, M. Pantic. A multimodal database for affect recognition and implicit tagging. *IEEE Transactions on Affective Computing*, 3(1):42–55, 2012.
- [18] J. Wijsman, B. Grundlehner, J. Penders, H. Hermens. Trapezius muscle EMG as predictor of mental stress. In: *Wireless Health 2010*, pp. 155–163, 2010.
- [19] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P. Patel-Schneider. T. Cambridge University Press, 2003.
- [20] M. Ruta, F. Scioscia, E. Di Sciascio. Concept Abduction and Contraction in Semantic-based P2P Environments, *Web Intelligence and Agent Systems*, 9(3), 179-207, 2011.
- [21] J. R. Parker. Rank and response combination from confusion matrix data. *Information fusion*, 2(2): 113-120, 2001.
- [22] K. Chomboon, P. Chujai, P. Teerarassamee, K. Kerdprasop, N. Kerdprasop. An empirical study of distance metrics for k-nearest neighbor algorithm. In: *Proceedings of the 3rd International Conference on Industrial Application Engineering*, 2015.
- [23] M. A. Musen. The Protégé project: A look back and a look forward. *AI Matters. Association of Computing Machinery Specific Interest Group in Artificial Intelligence*, 1(4), 2015.
- [24] P. Ekman, E. L. Rosenberg. *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, 1997.

## Ringraziamenti

Ci tengo a ringraziare la mia famiglia, per avermi sempre supportato, anche con il semplice interesse e la curiosità, in tutto il percorso di lavoro di tesi così come il mio intero percorso universitario. Il modo con cui mi hanno insegnato a guardare il lato positivo di ogni cosa, e ad affrontarla con la perseveranza di chi fa ciò che gli piace a prescindere dall'esito.

Vorrei anche ringraziare i miei amici più stretti con cui ho sempre condiviso le mie idee e le incertezze comuni, alleviando soprattutto il peso psicologico che la carriera universitaria ed ingegneristica può assumere in situazioni non sempre facili.

Ringrazio anche il professore Michele Ruta che mi ha dato l'opportunità di condurre il lavoro di tesi su di un argomento che ha sempre suscitato in me interesse, a partire dal punto di vista prettamente scientifico e tecnologico (incontrando perfettamente i miei interessi personali sia sul puro filo speculativo fantascientifico, che sulle mie personali attività hobbistiche in ambito tecnologico e dell'automazione), fino alle implicazioni benefiche sociali ed umane che un mezzo di tale calibro, con le giuste precauzioni e nelle giuste mani, può portare in contesti come la sanità pubblica, l'assistenza, il miglioramento e la consapevolezza approfondita delle condizioni fisiche e psicologiche dell'individuo.

Un grosso ringraziamento va agli ingegneri Floriano Scioscia, Agnese Pinto e Giuseppe Loseto, che mi hanno sempre supportato durante tutto il lavoro di tesi aiutandomi sia dal punto di vista pratico, risolvendo eventuali dubbi o perplessità al massimo delle loro possibilità e con la totale disponibilità, che dal punto di vista comunicativo, sempre riconoscendo e dando peso ai risultati ottenuti qualsiasi questi fossero, incoraggiandomi direttamente ed indirettamente al progressivo miglioramento.