



UNIVERSITÀ DI PISA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Corso di laurea in Ingegneria informatica

IA NELL'AMBITO DELL'AFFECTIVE COMPUTING

Candidato: Gianluca Gemini

Relatori: Prof. Beatrice Lazzerini

Dr. Francesco Pistolesi

Anno accademico 2020-2021

INDICE

1. INTRODUZIONE	1
1.1. Utilità in campo medico	1
1.2. Utilità sociale	2
1.3. Incidenti sul lavoro	3
2. BACKGROUND	4
2.1. Reti Neurali	4
2.2. Tipi di Machine Learning	7
2.3. Alberi di decisione	8
3. REGISTRAZIONI CON MUSE E SHIMMER	11
3.1. Introduzione ai dispositivi utilizzati	11
3.2. Registrazioni effettuate	13
3.3. Risultati dopo l'addestramento	15
4. STUDIO ED ELABORAZIONE DEI DATI DI DEAP	17
4.1. Introduzione a DEAP	17
4.2. Premesse	19
4.3. Estrazione delle features	19
4.4. Identificazione delle emozioni	21
4.5. Confronto con i grafici	22
4.6. Analisi approfondita dei grafici	25
5. CONCLUSIONI FINALI	28
BIBLIOGRAFIA	29

1. INTRODUZIONE

La tesi riguarda l'addestramento di un sistema di intelligenza artificiale applicato alla rilevazione delle emozioni umane tramite dei dispositivi indossabili. I dati raccolti da questi dispositivi vengono elaborati tramite la tecnica del machine learning, in particolare dell'albero di decisione (Classificatore), e l'output classificato in una delle principali emozioni umane tramite un'emoji. Il Classificatore in questione è stato pre-addestrato su un piccolo set di registrazioni effettuate su una decina di persone, l'obiettivo è quello arricchire il set di addestramento per migliorarne le prestazioni e valutarne i cambiamenti. Prima di parlare dell'argomento vero e proprio è opportuno trattare brevemente i campi di applicazione in cui un dispositivo di questo genere si collocherebbe, come per esempio il campo medico o quello di previsione di incidenti sul lavoro.

1.1. Utilità in campo medico

Per quanto riguarda il campo medico può essere un prezioso strumento per monitorare la salute psicofisica dei pazienti in cura, fornirebbe uno storico completo delle emozioni rilevate durante le giornate, quindi aiutando il medico nell'individuazione di un eventuale disagio, disturbo o patologia psichica. Infatti, in questo campo è frequente l'uso di tecniche di neuroimaging (mappatura diretta o indiretta della struttura, della funzione o della farmacologia del sistema nervoso) che combinate alla rilevazione delle emozioni permetterebbe una visione più profonda della psiche del paziente.

1.2. Utilità sociale

Tale tecnologia potrebbe sventare o prevenire con largo anticipo i casi di suicidio, omicidio o violenza. Infatti, il dispositivo avviserebbe il soggetto già dai primi segnali su emozioni pericolose quali eccessivo stress, depressione o frequenti episodi di rabbia, e qualora il soggetto fosse in cura o avesse precedenti penali, accuse di stalking e violenza, i dati raccolti e i risultati di allerta potrebbero essere condivisi con il medico curante o con le forze dell'ordine.

Un recente fatto di cronaca nera che si sarebbe potuto evitare in questo modo è il caso di Vanessa, siciliana, uccisa ad Aci Trezza nella notte tra il 22 e il 23 agosto dall'ex fidanzato Antonio a colpi di pistola, dopo di che si sarebbe suicidato. Antonio era già stato accusato di stalking e condannato agli arresti domiciliari, in seguito, una volta libero, era stata emessa la misura cautelare per divieto di avvicinamento. In questo particolare caso si sarebbe potuto applicare al ragazzo il dispositivo dopo la condanna di stalking e questo avrebbe sicuramente rilevato in tempo utile la sua condizione emotiva prima dell'omicidio, e avvertito le autorità che sarebbero intervenute salvando la vita di Vanessa. Come in questo caso il dispositivo potrebbe scongiurare episodi di violenza sulle donne e femminicidio, secondo i dati dell'Istat si stima che il 21,5% delle donne tra i 16 e i 70 anni abbia subito nel corso della propria vita episodi di stalking e il 31% una qualunque forma di violenza fisica o sessuale. Inoltre, il 61,3% delle donne uccise nel 2019 sono state assassinate dal partner attuale o ex, il 22,5% da un parente.

Il dispositivo potrebbe evitare che individui particolarmente vulnerabili compiano gesti estremi, in Italia si stimano secondo i dati dell'Istat circa 4000 suicidi di cui il 5% sono giovani sotto i 24 anni. Il suicidio nei giovani studenti è un tema molto sentito, le principali cause sono il bullismo e il cyberbullismo per i più giovani come nei casi di una 13enne di Roma, una 13enne di Varese e un 17enne di Rivoli, e i fallimenti in ambito universitario per la fascia di età oltre i 19 anni. Tra questi ultimi tragici eventi pre-covid19 possiamo citare un 25enne del corso di Medicina a Pavia nel 2019, una 26enne di Medicina a Perugia, un 20enne di Ingegneria a Genova, ma ci sono casi anche di ragazzi del liceo come un 18enne e

una 19enne del liceo scientifico Frisi di Monza, e un 18enne del liceo classico Monti di Cesena. Secondo l'ospedale romano Bambino Gesù negli ultimi otto anni le richieste urgenti per tentati suicidi sono aumentate di venti volte dal 2017 al 2018, mentre le consulenze neuropsichiatriche sono aumentate del 24% rispetto al 2017.

È chiaro che il monitoraggio delle emozioni può essere applicato, salvo su richiesta di un medico, solo in seguito al primo evento di tentato suicidio o violenza, altrimenti i suoi dati non potrebbero essere condivisi con altri per questioni di privacy che però non è oggetto di discussione.

1.3. Incidenti sul lavoro

Come accennato il sistema di riconoscimento può essere sfruttato per evitare gli incidenti sul lavoro dovuti a disattenzione o improvvisa insorgenza di malori. Secondo i dati Inail gli incidenti denunciati nel 2019 sono stati oltre 640 mila, la fascia di età più colpita è quella tra i 40 e i 60 anni, le modalità di accadimento sono nella stragrande maggioranza senza mezzi di trasporto in occasione di lavoro e con mezzi di trasporto per incidenti in itinere. La rilevazione delle emozioni potrebbe abbattere questi dati, infatti le emozioni più forti come rabbia, depressione o anche stati d'animo anomali causati dall'abuso di sostanze o da patologie, spesso sono incompatibili con il livello di attenzione richiesto per svolgere alcuni lavori. Dunque, ci si potrebbe avvalere di tale strumento per avvisare i lavoratori stessi di fare attenzione o in casi eccezionali di notificare l'incompatibilità dell'individuo con il lavoro da svolgere. Inoltre, a seguito di un incidente potrebbe essere utile per identificare la responsabilità dei lavoratori e migliorare i sistemi di sicurezza.

2. BACKGROUND

Come accennato la tecnica utilizzata per il riconoscimento delle emozioni è il machine learning, è una categoria di algoritmi che si migliorano man mano che vengono utilizzati, per questo è necessario un periodo di addestramento con lo scopo di rendere questi algoritmi efficienti. L'idea alla base del machine learning è la simulazione di un cervello umano o il suo modo di pensare, ovviamente molto semplificato, in genere in grado di raggiungere uno o pochi obiettivi, per questo all'inizio non è in grado di svolgere il proprio compito ma impara a farlo piano piano proprio come un cervello umano. Due dei principali modelli per il machine learning sono le Reti Neurali e l'Albero di decisione.

2.1. Reti Neurali

L'elemento chiave è il Perceptron, un'astrazione che prende diversi input binari x_i tramite dei canali di ingresso, ogni canale ha un peso w_i . Il Perceptron produrrà un solo output y binario e sarà 1 se il prodotto scalare tra il vettore degli input e il vettore dei pesi sarà superiore ad una certa soglia, altrimenti 0. Normalmente la soglia viene spostata al di là dell'uguale e chiamata bias (b).

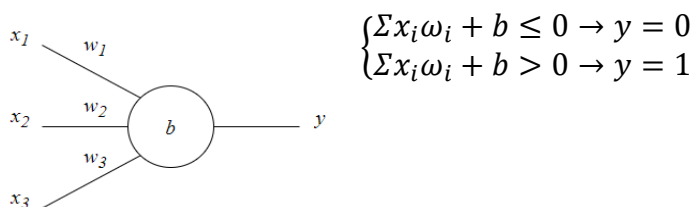


Figura 1. Un esempio di Perceptron con regola matematica

Il problema del Perceptron è che una piccola modifica dei pesi e del bias potrebbe cambiare l'output troppo facilmente, per questo si usa una sua variante detta

Neurone Sigmoidale. La differenza sta nell'output che viene ottenuto dando in input il prodotto scalare sommato con il bias ad una funzione detta di attivazione, il cui output varia da 0 a 1 in R e fa in modo che piccole variazioni dei pesi e del bias provochino piccole variazioni dell'output. Un esempio di funzione di attivazione è la cosiddetta Sigmoidale.

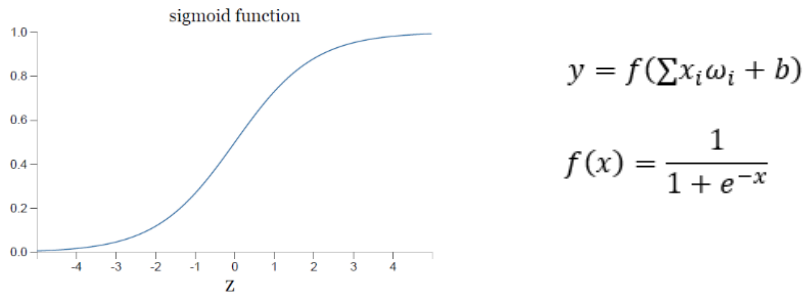


Figura 2. Grafico e formula della funzione Sigmoidale, output (y) del neurone sigmoidale

Possiamo vedere il Perceptron come un Neurone Sigmoidale che ha come funzione di attivazione la funzione gradino. Ciò che ci permette di costruire un algoritmo in grado di imparare da solo è una Rete Neurale che è una rete di Neuroni Sigmoidali su più livelli.

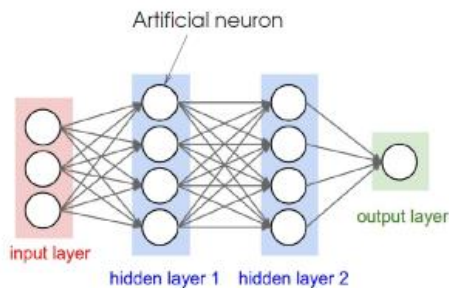


Figura 3. Esempio di Rete Neurale

I neuroni del primo livello non sono veri e propri neuroni ma solo una specie di generatori di input per il prossimo livello. Ogni neurone produce un solo output che distribuisce a tutti i neuroni del livello successivo e riceve tanti input quanti sono i neuroni del livello precedente. Il livello di output è

costituito da un solo neurone. Questa struttura aciclica è detta Rete Feedforward, tuttavia anche se meno usate si possono utilizzare strutture con circuiti di feedback dette Ricorrenti. Una rete con più di un livello nascosto (vedi figura 3) è detta MLP (Multilayer Perceptron), se ogni livello ne nasconde altri la rete è detta Deep altrimenti Shallow.

La caratteristica sorprendente del machine learning è il motivo per cui ha trovato largo impiego, è la capacità di imparare. Per prima cosa si definisce una funzione che faccia una stima dell'errore della predizione (output), per esempio se il compito dell'algoritmo è riconoscere una lettera dell'alfabeto la funzione indicherà la percentuale di fallimento. Ovviamente l'obiettivo è minimizzare questa percentuale con lo scopo di rendere l'algoritmo idealmente infallibile.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_{true} - y_{pred})^2$$

Un esempio di funzione dell'errore è la MSE (Mean Square Error), dove n è il numero di casi, y_{true} è la predizione corretta e y_{pred} è la predizione calcolata dall'algoritmo. Per esempio, supponiamo di avere quattro persone adulte di cui conosciamo peso e altezza (input) e la macchina deve indovinare il genere (output). Codifichiamo con 0 il genere femminile e con 1 il genere maschile, ipotizziamo che le persone siano nell'ordine maschio, femmina, femmina, maschio (1, 0, 0, 1) e che l'algoritmo calcoli le seguenti predizioni 0, 0, 0, 0 (tutte femmine).

$$MSE = \frac{1}{4} ((1 - 0)^2 + (0 - 0)^2 + (0 - 0)^2 + (1 - 0)^2) = \frac{2}{4} = \frac{1}{2} = 50\%$$

Secondo il nostro esempio l'algoritmo indovina il 50% delle volte, ovviamente per calcolare correttamente l'MSE è necessario un numero di casi molto più grande. Ricordando che y_{pred} è l'output della Rete Neurale e dunque dell'ultimo Neurone Sigmoidale, questo è dovuto agli output calcolati da ogni neurone tramite la funzione Sigmoidale, pesi e bias. Quindi per minimizzare l'MSE bisogna intervenire modificando i pesi e i bias della rete neurale e per capire come, si usa un algoritmo detto Stochastic Gradient Descent. All'inizio i pesi e i bias sono assegnati casualmente, poi durante un periodo detto di 'training' si sottopone alla rete un dataset contenente molte coppie costituite ognuna dall'input e dal corrispondente output desiderato, l'algoritmo di discesa stocastica del gradiente si occuperà di modificare i bias e i pesi in modo da minimizzare l'errore.

L'algoritmo in questione è un metodo iterativo per l'ottimizzazione di funzioni differenziali, quindi è necessario che il dataset sia sufficientemente grande. Dopo aver sottoposto l'intero dataset si dice che si abbia concluso un'epoca di addestramento, dopo di che si ricomincia da capo con lo stesso dataset per la seconda epoca, poi la terza e così via fino al raggiungimento di una percentuale di successo accettabile.

2.2. Tipi di Machine Learning

Ci sono diversi tipi di machine learning, i più utilizzati sono il Supervised Learning, l'Unsupervised Learning, il Generative Adversarial Network e il Reinforcement Learning.

Il primo utilizza un sistema di etichette dette 'label' che vengono assegnati agli input, tale assegnazione è proprio l'output, durante l'addestramento vengono fornite alla rete coppie input-label. Se il label è un valore numerico allora si parla di Regressione altrimenti di Classificazione. Un esempio di questa tipologia è la classificazione delle mail in SPAM e i film consigliati sulle piattaforme di streaming (sfruttano vari label come genere, target e durata). Questo è il metodo più usato, tuttavia è necessario un dataset molto grande durante la fase di addestramento.

Il secondo tipo è più complicato perché non sfrutta le etichette ma in compenso non ha bisogno di un dataset per la fase di training, l'obiettivo della rete è raggruppare gli input in classi (Clustering) secondo caratteristiche comuni o trovare associazioni tra loro (Association). Questo metodo viene usato per trovare anomalie tra gli input come, per esempio, transazioni fraudolente per conto delle banche.

Il Generative Adversarial Network è in realtà due Reti Neurali che hanno obiettivi opposti, una è detta Generator che produce la previsione e l'altra detta Discriminator che la contesta, il punto forte di questo metodo è la continua lotta tra le due reti che si migliorano a vicenda.

L'ultimo, il Reinforcement Learning, utilizza un sistema ispirato al metodo di apprendimento degli esseri viventi, infatti utilizza delle ricompense che possono essere banalmente degli incrementi ad una variabile o decrementi in caso di penalità. La rete, durante il periodo di addestramento, è chiamata a prendere delle decisioni che vengono poi premiate se corrette o penalizzate se sbagliate. L'obiettivo della rete è ottenere ricompense.

2.3. Alberi di decisione

Un albero di decisione è un modello di machine learning molto diverso dalle Reti Neurali, è di tipo Supervised Learning e quindi usato per la Classificazione o per la Regressione.

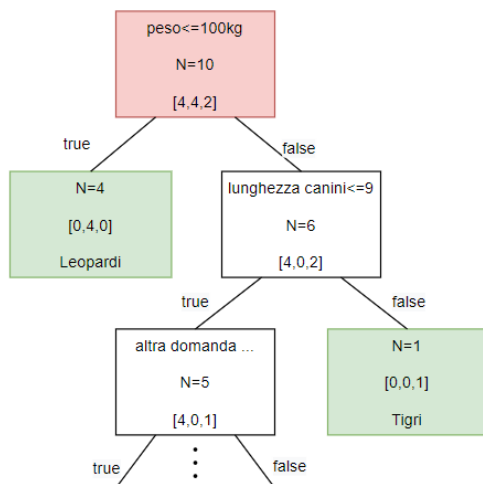


Figura 4. Esempio di albero di decisione

	peso(kg)	lunghezza(m)	l. canini(cm)	tipo
1	30	1,6	7	Leopardo
2	110	2	8	Tigre
3	28	1	6	Leopardo
4	128	1,7	8	Leone
5	131	2,1	8	Leone
6	27	1,6	7	Leopardo
7	31	1,5	6	Leopardo
8	220	3	10	Tigre
9	180	2	7	Leone
10	175	1,9	7	Leone

Figura 5. Dataset di training

Nella figura 4 è mostrato un esempio di albero di decisione che ha il compito di distinguere 10 animali in leopardi, leoni e tigri, i cui dati si trovano in un dataset (figura 5) organizzati in caratteristiche dette 'features' considerate importanti per la decisione (peso, lunghezza del corpo, lunghezza dei canini). Per capire come funziona prendiamo il nodo radice in rosso, questo è composto da una domanda su una particolare feature (peso) e un valore detto 'punto di divisione' (100), 'N' indica il numero di elementi (10 animali) a cui bisogna applicare la domanda, e il

‘vettore delle classi’ [4,4,2] indica che ci sono nell’ordine 4 leoni, 4 leopardi e 2 tigri.

tipo	peso	<=100
Leopardo	30	True
Tigre	110	False
Leopardo	28	True
Leone	128	False
Leone	131	False
Leopardo	27	True
Leopardo	31	True
Tigre	220	False
Leone	180	False
Leone	175	False

Figura 6

tipo	L.canini	<=9
Tigre	8	True
Leone	8	True
Leone	8	True
Tigre	10	False
Leone	7	True
Leone	7	True

Figura 7

Tutto quello che si è appena visto accade durante la fase di training, il dataset mostrato in figura 5 infatti comprende anche i label come è tipico nel Supervised Learning, dopo l’addestramento l’albero sarà in grado di assegnare la classe corretta sottoponendo le feature ai vari nodi fino a raggiungere una foglia. La struttura dell’albero si genera automaticamente durante la fase di addestramento e rimarrà la stessa dopo la sua conclusione. Per capire come avviene la sua generazione si prenda lo stesso esempio fatto in precedenza e si ipotizzi di non avere ancora un nodo radice. Bisogna scegliere una feature e un punto di divisione con cui fare la domanda, pare fare questo si usa una funzione chiamata ‘Guadagno di Informazione’ (IG) e uno tra i due criteri detti ‘criterio di Gini’ (I_G) e ‘criterio dell’entropia’ (I_H).

Come mostrato nella tabella in figura 6 quattro dei dieci animali hanno risposto ‘true’ alla domanda nel nodo rosso, si formano così due sottogruppi: il primo costituito dagli animali con il peso minore o uguale a 100 e il secondo costituito dai restanti. I due gruppi vengono assegnati ciascuno ai nodi sottostanti detti figli (quello rosso è detto padre). Ad ogni decisione si scende di un livello e il numero massimo di livelli è detto profondità dell’albero. Notiamo che il figlio sinistro di livello due contiene animali tutti dello stesso tipo [0,4,0] (leopardi), quindi non sono necessarie ulteriori divisioni, tale nodo è detto foglia o nodo puro. Stessa cosa vale per il nodo di livello tre generatosi dopo il punto di divisione sulla lunghezza dei canini, i 6 animali (4 leoni e 2 tigri) rispondono alla domanda come nella tabella in figura 7. Il nodo sinistro di livello tre dovrà dividersi ulteriormente perché contiene due gruppi differenti (4 leoni e 1 tigre).

$$IG(D_P) = I(D_P) - \frac{N_L}{N_P} I(D_L) - \frac{N_R}{N_P} I(D_R)$$

$$I(D_P) = I_G(D_P) = 1 - \sum_{i=1}^C P_i^2, \quad P_i = \frac{N_i}{N_P}$$

$$I(D_P) = I_H(D_P) = - \sum_{i=1}^C P_i \log_2(P_i)$$

D_P = dataset del padre

D_R = dataset del figlio destro

D_L = dataset del figlio sinistro

N_P = numero elementi del padre

N_R = numero elementi del figlio destro

N_L = numero elementi del figlio sinistro

N_i = valore i-esimo del vettore delle classi

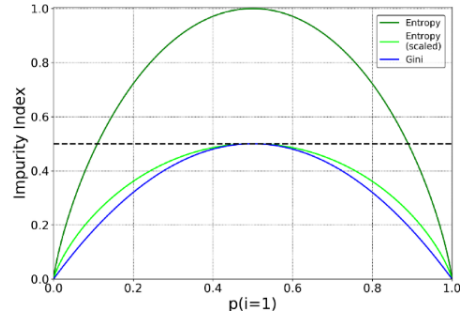
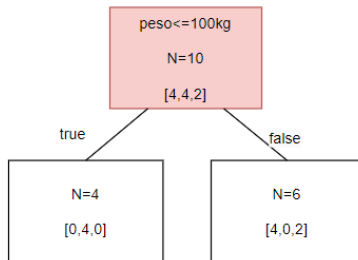


Figura 8. Grafico dei due criteri

È preferibile il criterio di Gini perché è più facile da calcolare ed è comunque molto simile al criterio dell'entropia come mostrato in figura 8. Tornando alla scelta della domanda, un algoritmo prova per ogni feature e ogni punto di divisione (in un intervallo consentito) quale massimizza IG , di seguito vi è un esempio del calcolo di IG usando come feature il peso e come punto di divisione 100, vengono costruiti i figli seguendo la tabella della figura 6 e si cominciano a calcolare i valori (figura 9).



$$I_G(D_P) = 1 - \left(\left(\frac{4}{10} \right)^2 + \left(\frac{4}{10} \right)^2 + \left(\frac{2}{10} \right)^2 \right) = 0.64$$

$$I_G(D_L) = 1 - \left(\left(\frac{0}{4} \right)^2 + \left(\frac{4}{4} \right)^2 + \left(\frac{0}{4} \right)^2 \right) = 0$$

$$I_G(D_R) = 1 - \left(\left(\frac{4}{6} \right)^2 + \left(\frac{0}{6} \right)^2 + \left(\frac{2}{6} \right)^2 \right) = 0.44$$

$$IG(D_P) = 0.64 - \left(\frac{4}{10} \right) * 0 - \left(\frac{6}{10} \right) * 0.44 = 0.37$$

Figura 9. Costruzione temporanea del nodo padre e dei nodi figli, verrà confermata solo se IG è il migliore

Una volta trovata la migliore feature e il migliore punto di divisione, l'algoritmo riparte da capo sui nodi figli non puri costruendo nuove domande e nuovi figli. Questo processo non si conclude finché tutti i nodi terminali non sono foglie e ciò potrebbe portare ad un albero con una profondità eccessiva, per questo si limita l'algoritmo ad una profondità massima. È bene notare che ogni volta che riparte l'algoritmo su un nodo figlio diminuisce il numero degli elementi N e cambia il vettore delle classi.

3. REGISTRAZIONI CON MUSE E SHIMMER

In questa prima parte ci si è occupati di registrare i canali EEG (elettroencefalogramma), PPG (fotopletismografia, valutazione del flusso sanguigno), GSR (risposta galvanica della pelle) e EMG (elettromiografia) tramite il dispositivo Muse-S, lo Shimmer3 EMG e lo Shimmer3 GSR+, con lo scopo di arricchire il dataset di addestramento per un Classificatore, riaddestrarlo e di valutarlo su nuovi soggetti. Tutti questi dispositivi campionano a 256 Hz e sono collegati via bluetooth ad uno smartphone Android che li gestisce tramite un'app chiamata SmartEmotions.

3.1. Introduzione ai dispositivi utilizzati

Il Muse-S è un dispositivo capace di misurare i canali TP9, AF7, AF8, e TP10 degli EEG, va indossato come una semplice fascia sportiva, è costituito da una parte in plastica che deve essere posta sulla fronte e una parte elastica che deve circondare il cranio passando dietro le orecchie.



Figura 10. Muse-S

I due Shimmer sono dispositivi indossabili tramite delle cinghie elastiche e presentano dei connettori a cui collegare degli elettrodi per le misurazioni. Lo Schimmer3 GSR+ ha due ingressi per la risposta galvanica della pelle, gli elettrodi vanno applicati tramite due fasce circolari uno all'indice e un al medio, e ha una terza entrata per la fotopletismografia il cui elettrodo va applicato all'anulare. Lo Shimmer3 EMG ha 5 ingressi per l'elettromiografia, gli elettrodi sono stati posizionati come in figura 12.



Figura 11. Shimmer3 GSR+

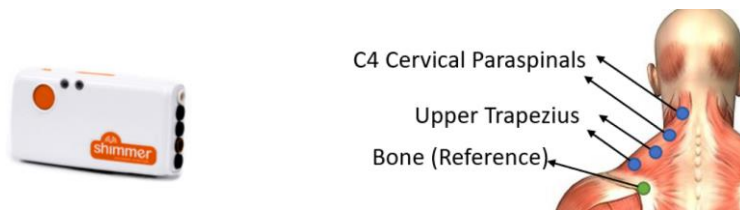


Figura 12. Shimmer3 EMG

Per quanto riguarda l'app, questa viene avviata in background e si attiva dopo un tempo randomico nell'ordine di una decina di minuti in modo che chi partecipa alle registrazioni non sappia il momento esatto in cui avvengano. Dopo una registrazione di durata 2 minuti, l'app invia una notifica per chiedere all'utente quale emozione ha provato tramite delle emoji rappresentanti rabbia, ansia, noia, disgusto, paura, felicità e tristezza, ognuna da un livello da 1 a 3, in fine i dati vengono raccolti in un server (i livelli delle emoji non sono stati considerati).



Figura 13. Emoji per l'autovalutazione nell'app, ogni emoji ha tre livelli

3.2. Registrazioni effettuate

Di seguito vi è la tabella delle registrazioni che sono state effettuate, ogni registrazione è stata divisa in finestre da 10 secondi, i dati sono stati raccolti tramite uno script in Matlab che ne ha calcolato le features. Infine le features sono state date in input al classificatore che ha ottenuto i livelli di accuratezza mostrati in tabella per la previsione delle emozioni.

ID	Genere	Età	Numero di registrazioni	Attività svolta	Accuratezza
1	Donna	25	7	Videogioco horror “Resident Evil VII”	5.3%
2	Uomo	23	8	Videogioco d’azione “Doom Eternal”	13.2%
3	Donna	22	5	Aperitivo e camminata	20.0%
4	Donna	22	3	Chiacchiere tra amici	0%
5	Uomo	23	9	partecipazione ad una riunione online	0%
6	Uomo	23	6	studio	3.4%
7	Uomo	23	7	studio	23.5%
8	Uomo	73	6	Pranzo di famiglia	51.7%

		Confusion Matrix							
Output Class	anger	1 0.4%	0 0.0%	0 0.0%	2 0.8%	0 0.0%	0 0.0%	0 0.0%	33.3% 66.7%
	happiness	31 12.1%	23 9.0%	13 5.1%	87 34.0%	13 5.1%	6 2.3%	0 0.0%	13.3% 86.7%
	sadness	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	NaN% NaN%
	anxiety	7 2.7%	10 3.9%	0 0.0%	5 2.0%	0 0.0%	7 2.7%	0 0.0%	17.2% 82.8%
	fear	1 0.4%	15 5.9%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0.0% 100%
	boredom	2 0.8%	8 3.1%	0 0.0%	11 4.3%	0 0.0%	6 2.3%	0 0.0%	22.2% 77.8%
	disgust	0 0.0%	0 0.0%	0 0.0%	8 3.1%	0 0.0%	0 0.0%	0 0.0%	0.0% 100%
		2.4% 97.6%	41.1% 58.9%	0.0% 100%	4.4% 95.6%	0.0% 100%	31.6% 68.4%	NaN% NaN%	13.7% 86.3%
		Target Class							
		anger	happiness	sadness	anxiety	fear	boredom	disgust	

Figura 14. Matrice di confusione che indica le percentuali di successo (in verde) per tutte le registrazioni

La matrice di confusione in figura 25 racchiude tutte le finestre di registrazione, presenta per colonne le emozioni indicate dai partecipanti tramite l'app e per righe le emozioni calcolate dall'albero di decisione, nella diagonale in verde ci sono le emozioni classificate correttamente. I numeri nelle caselle rosse indicano le finestre con un'emozione non riconosciuta. Osservando la diagonale si può notare come il Classificatore abbia previsto almeno una volta tutte le emozioni tranne la paura e la tristezza, il disgusto non è mai stato indicato dalle autovalutazioni dei partecipanti e mai previsto dall'albero (colonna di zeri). È evidente la tendenza a confondere la rabbia e l'ansia con la felicità.

3.3. Risultati dopo l'addestramento

I dati registrati sono stati accodati al dataset originale con cui il Classificatore era stato addestrato e sui quali aveva ottenuto il 100% di accuratezza, poi si è usato questo nuovo database per addestrarne uno nuovo con Matlab. Dopo di che il nuovo albero decisionale è stato testato sia con i vecchi dati sia con le nuove registrazioni. Nella tabella a seguire è mostrato la percentuale di accuratezza raggiunta per i soggetti (visti prima) delle nuove registrazioni dopo l'addestramento.

ID	1	2	3	4	5	6	7	8
Accuratezza	97.4%	94.3%	100%	91.7%	100%	93.1%	91.2%	100%

Confusion Matrix									
Output Class	anger	63 7.4%	5 0.6%	4 0.5%	0 0.0%	0 0.0%	3 0.4%	0 0.0%	84.0% 16.0%
	happiness	0 0.0%	232 27.3%	1 0.1%	0 0.0%	0 0.0%	12 1.4%	0 0.0%	94.7% 5.3%
	sadness	0 0.0%	1 0.1%	52 6.1%	0 0.0%	0 0.0%	3 0.4%	0 0.0%	92.9% 7.1%
	anxiety	0 0.0%	24 2.8%	7 0.8%	233 27.4%	0 0.0%	26 3.1%	0 0.0%	80.3% 19.7%
	fear	0 0.0%	4 0.5%	0 0.0%	0 0.0%	29 3.4%	3 0.4%	0 0.0%	80.6% 19.4%
	boredom	0 0.0%	7 0.8%	1 0.1%	0 0.0%	0 0.0%	126 14.8%	0 0.0%	94.0% 6.0%
	disgust	0 0.0%	1 0.1%	1 0.1%	0 0.0%	0 0.0%	2 0.2%	10 1.2%	71.4% 28.6%
		100% 0.0%	84.7% 15.3%	78.8% 21.2%	100% 0.0%	100% 0.0%	72.0% 28.0%	100% 0.0%	87.6% 12.4%
		Target Class							
		anger	happiness	sadness	anxiety	fear	boredom	disgust	

Figura 15. Matrice di confusione sull'intero dataset di training

		Confusion Matrix						
Output Class	anger	42 16.4%	2 0.8%	1 0.4%	0 0.0%	0 0.0%	0 0.0%	93.3% 6.7%
	happiness	0 0.0%	52 20.3%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	sadness	0 0.0%	0 0.0%	10 3.9%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	anxiety	0 0.0%	1 0.4%	2 0.8%	113 44.1%	0 0.0%	3 1.2%	95.0% 5.0%
	fear	0 0.0%	1 0.4%	0 0.0%	0 0.0%	13 5.1%	0 0.0%	92.9% 7.1%
	boredom	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	16 6.2%	100% 0.0%
	100% 0.0%	92.9% 7.1%	76.9% 23.1%	100% 0.0%	100% 0.0%	84.2% 15.8%	96.1% 3.9%	
		anger	happiness	sadness	anxiety	fear	boredom	
		Target Class						

Figura 16. Matrice di confusione per le nuove registrazioni, il disgusto (non presente) non è stato mai né calcolato né indicato dai soggetti

		Confusion Matrix							
Output Class	anger	21 3.5%	0 0.0%	3 0.5%	0 0.0%	0 0.0%	3 0.5%	3 0.5%	70.0% 30.0%
	anxiety	0 0.0%	120 20.2%	23 3.9%	0 0.0%	0 0.0%	23 3.9%	5 0.8%	70.2% 29.8%
	boredom	0 0.0%	0 0.0%	110 18.5%	0 0.0%	0 0.0%	7 1.2%	1 0.2%	93.2% 6.8%
	disgust	0 0.0%	0 0.0%	2 0.3%	10 1.7%	0 0.0%	1 0.2%	1 0.2%	71.4% 28.6%
	fear	0 0.0%	0 0.0%	3 0.5%	0 0.0%	16 2.7%	3 0.5%	0 0.0%	72.7% 27.3%
	happiness	0 0.0%	0 0.0%	12 2.0%	0 0.0%	0 0.0%	180 30.3%	1 0.2%	93.3% 6.7%
	sadness	0 0.0%	0 0.0%	3 0.5%	0 0.0%	0 0.0%	1 0.2%	42 7.1%	91.3% 8.7%
	100% 0.0%	100% 0.0%	70.5% 29.5%	100% 0.0%	100% 0.0%	82.6% 17.4%	79.2% 20.8%	84.0% 16.0%	
		anger	anxiety	boredom	disgust	fear	happiness	sadness	
		Target Class							

Figura 17. Matrice di confusione per il dataset originale

Il nuovo Classificatore risulta notevolmente migliorato sui nuovi dati, infatti è passato dal 13.7% (figura 14) ad un 96.1% (figura 16), mentre l'accuratezza sul dataset originale è scesa dal 100% all'87%, in generale l'albero di decisione ha raggiunto buoni risultati imparando a riconoscere le emozioni di nuovi soggetti.

4. STUDIO ED ELABORAZIONE DEI DATI DI DEAP

In questa seconda parte ci si è occupati di valutare le previsioni effettuate dal nuovo Classificatore sul dataset DEAP e il sistema di autovalutazione utilizzato durante la sua costruzione. Questo dataset è stato pubblicato nel 2012 con l'articolo 'DEAP: A Database for Emotion Analysis using Physiological Signals' e messo a disposizione dei ricercatori per utilizzarlo in ambito dell'affective computing.

4.1. Introduzione a DEAP

È stato prodotto sottoponendo 32 volontari all'ascolto di 40 video musicali di un minuto ciascuno, opportunamente selezionati. Durante l'ascolto sono stati registrati 40 segnali, 8064 campioni per ognuno, rilevati in 63 secondi con sottocampionamento a 128Hz. Di questi, 32 sono canali EEG e gli altri 8 sono i seguenti: due canali per gli EOG (elettrooculogramma), due canali per gli EMG, un canale per il GSR, uno per il Respiration belt (cambiamenti della circonferenza toracica o addominale durante la respirazione), uno per la PPG e uno per la temperatura corporea.

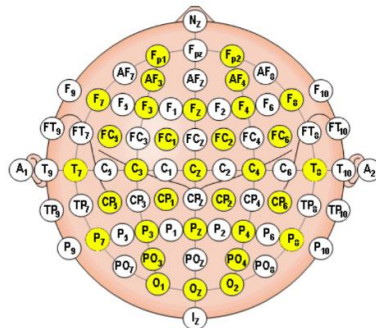


Figura 18. In evidenza i canali EEG registrati

33	hEOG (horizontal EOG, hEOG ₁ - hEOG ₂)
34	vEOG (vertical EOG, vEOG ₁ - vEOG ₂)
35	zEMG (Zygomaticus Major EMG, zEMG ₁ - zEMG ₂)
36	tEMG (Trapezius EMG, tEMG ₁ - tEMG ₂)
37	GSR (values from Twente converted to Geneva format (Ohm))
38	Respiration belt
39	Plethysmograph
40	Temperature

Figura 19. Tabella degli 8 canali non EEG

Channel name	Channel content
EXG1	hEOG ₁ (to the left of left eye)
EXG2	hEOG ₂ (to the right of right eye)
EXG3	vEOG ₁ (above right eye)
EXG4	vEOG ₄ (below right eye)
EXG5	zEMG ₁ (Zygomaticus Major, +/- 1cm from left corner of mouth)
EXG6	zEMG ₂ (Zygomaticus Major, +/- 1cm from zEMG ₁)
EXG7	tEMG ₁ (Trapezius, left shoulder blade)
EXG8	tEMG ₂ (Trapezius, +/- 1cm below tEMG ₁)
GSR1	Galvanic skin response, left middle and ring finger
GSR2	Unused

Figura 20. Approfondimento sui canali della figura 19

Al termine di ogni video tutti i partecipanti hanno rappresentato la propria emozione utilizzando delle scale discrete da 1 a 9 per misurare Valence (se si prova piacere o dispiacere), Arousal (eccitazione), Dominance (quanto si è dominati dall'emozione), liking (se è piaciuto il video) e una ruota delle emozioni (figura 21). Il dataset DEAP è organizzato in 32 file '.mat' (estensione Matlab), uno per ogni partecipante, contenenti ciascuno due tabelle: la prima chiamata 'data' di dimensione 40x40x8064 (video x sensori x campioni) e la seconda chiamata 'labels' di dimensione 40x4 contenente i valori di Valence, Arousal, Dominance e Liking per ogni video.

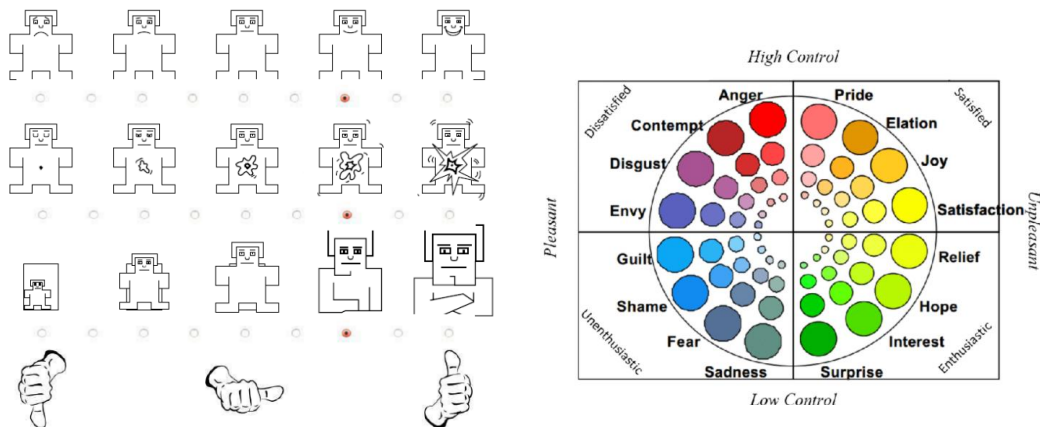


Figura 21. Scale discrete usate dai partecipanti durante l'esperimento DEAP e un esempio di ruota delle emozioni

4.2. Premesse

L'albero decisionale prende in ingresso delle features che sono state calcolate su alcuni dei segnali forniti da DEAP, in uscita produce per ogni persona e video un'emozione rappresentata tramite un'emoji. Dopo aver ottenuto le emozioni le si è confrontate con i valori di Valence e Arousal dichiarati dai partecipanti.

4.3. Estrazione delle features

Per estrarre le features innanzitutto sono state importate le 64 tabelle dei file '.mat' ('labels' e 'data', per ogni persona) del dataset DEAP, poi dalla tabella 'data' sono stati selezionati solo i canali compatibili con il Classificatore.

```

%%--ESTRAGGO I DATI DEI CANALI CHE MI INTERESSANO--
for k=1:sizeData
    EEGRAW(1,k)=(data(i,8,k)+data(i,12,k))/2; %RAW_TP9
    EEGRAW(2,k)=(data(i,1,k)+data(i,2,k)+data(i,3,k)+data(i,4,k))/4; %RAW_AF7
    EEGRAW(3,k)=(data(i,17,k)+data(i,18,k)+data(i,20,k)+data(i,21,k))/4; %RAW_AF8
    EEGRAW(4,k)=(data(i,26,k)+data(i,30,k))/2; %RAW_TP10
    EMG(k)=data(i,36,k);
    GSR(k)=data(i,37,k);
    PPG(k)=data(i,39,k);
end

```

Figura 22. Estratto di codice per il l'estrazione dei canali (sizeData=8064)

Come mostrato in figura 22, essendo necessari i canali TP9, AF7, AF8, e TP10, non registrati in DEAP (vedi figura 18), si è dovuto fare una media tra i canali presenti nel dataset, in particolare quelli evidenziati in figura 23. Inoltre sono stati selezionati i canali 36 per l'elettromiografia, 37 per la risposta galvanica della pelle e il 39 per la fotopletismografia (vedi figura 19).

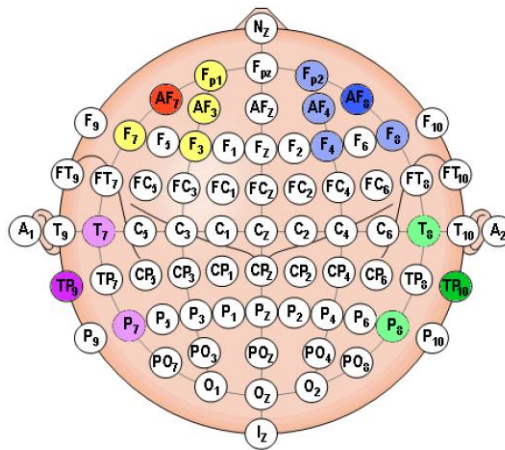


Figura 23. I colori più scuri rappresentano i canali mancanti, gli altri quelli usati per fare la media

Dopo di che sono state calcolate le features secondo quelle richieste dall'albero decisionale. In fine tutte le tabelle raggruppate per persona sono state unite e raggruppate per video, producendo un output di 80 tabelle in formato '.csv', due per ogni video, una di dimensione 32x20 (persone x features) e una di dimensione 32x2 contenente Valence e Arousal per ogni persona.

```
%calcolo features GSR
GSRFeatures = [median(GSR), max(GSR), rms(GSR), mean(GSR, 'omitnan')];
row=[row,GSRFeatures];
%calcolo features PPG
PPGFeatures = [median(PPG), min(PPG), max(PPG)];
row=[row,PPGFeatures];
```

Figura 24. Estratto di codice che mostra l'estrazione delle features dai canali GSR e PPG

```

for j = 1:4 %per ogni canale EEG

    % Calcolo features EEG
    if(j==1) %EEG1
        fftEEGRAW = fft(EEGRAW(j,:));
        betaWave = abs(ifft(bandpass(fftEEGRAW, [13 30], 128)));
        EEGFeatures = [median(EEGRAW(j,:)), obw(EEGRAW(j,:)), median(betaWave)];
    elseif(j==2) %EEG2
        fftEEGRAW = fft(EEGRAW(j,:));
        alphaWave = abs(ifft(bandpass(fftEEGRAW, [7.5 13], 128)));
        EEGFeatures = mean(alphaWave);
    elseif(j==3) %EEG3
        EEGFeatures = median(EEGRAW(j,:));
    elseif(j==4) %EEG4
        fftEEGRAW = fft(EEGRAW(j,:));
        thetaWave = abs(ifft(bandpass(fftEEGRAW, [4 7.5], 128)));
        EEGFeatures = [median(EEGRAW(j,:)), min(thetaWave)];
    end

    row=[row,EEGFeatures];

end

%calcolo features EMG
EMGFeatures = [median(EMG), max(EMG), min(EMG), trimmean(EMG, 10), mean(EMG, 'omitnan'), rms(EMG)];
row=[row,EMGFeatures];

```

Figura 25. Estratto di codice per l'estrazione delle feature dai canali EEG e EMG

4.4. Identificazione delle emozioni

Le tabelle prodotte nella fase precedente sono state utilizzate come input per questa fase. Ogni tabella contenente le features è stata sottoposta all'albero di decisione (chiamata alla funzione 'trainedModel.predictFcn(traccia)') che ha previsto per ogni persona e video l'emozione provata, in seguito le emozioni (rabbia, ansia, noia, disgusto, paura, felicità e tristezza) sono state salvate come stringhe accanto ai valori di Valence e Arousal visti in precedenza, producendo quindi una tabella 32x3 per ogni video.

```

nVideo=40;

for i=1:nVideo
    traccia=readtable(strcat("features/Features_t",num2str(i),".csv"));
    labels=readtable(strcat("features/Responses_t",num2str(i),".csv"));

    Emotion=trainedModel.predictFcn(traccia);
    Valance=labels(:,1);
    Arousal=labels(:,2);
    Emotion=array2table(Emotion);
    Results=[Valance,Arousal,Emotion];

    Results.Properties.VariableNames = {'Valence', 'Arousal','Emotion'};
    writetable(Results,strcat("results/Results_t",num2str(i),".csv"));
end

```

Figura 26. Estratto di codice che si occupa di ricavare le emozioni tramite il Classificatore

4.5. Confronto con i grafici

Nell'ultima fase sono stati costruiti 40 grafici per le registrazioni intere, uno per ogni video, con Arousal sulle ordinate e Valence sulle ascisse, inoltre ai punti è stata sostituita l'immagine dell'emoji calcolata. Dopo di che si è confrontato i grafici con un piano di Russell per verificare la corretta previsione e posizione delle emoji (sono state scelte le emoji di livello alto, vedi figura 13, per renderle più individuabili all'interno dei grafici).



Figura 27. Le elmoji usate: nell'ordine rabbia, ansia, noia, disgusto, paura, felicità e tristezza

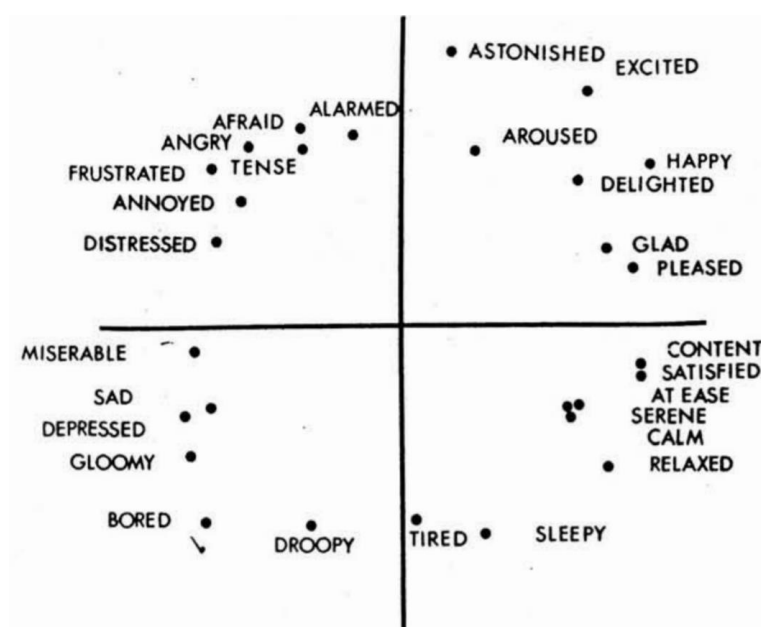
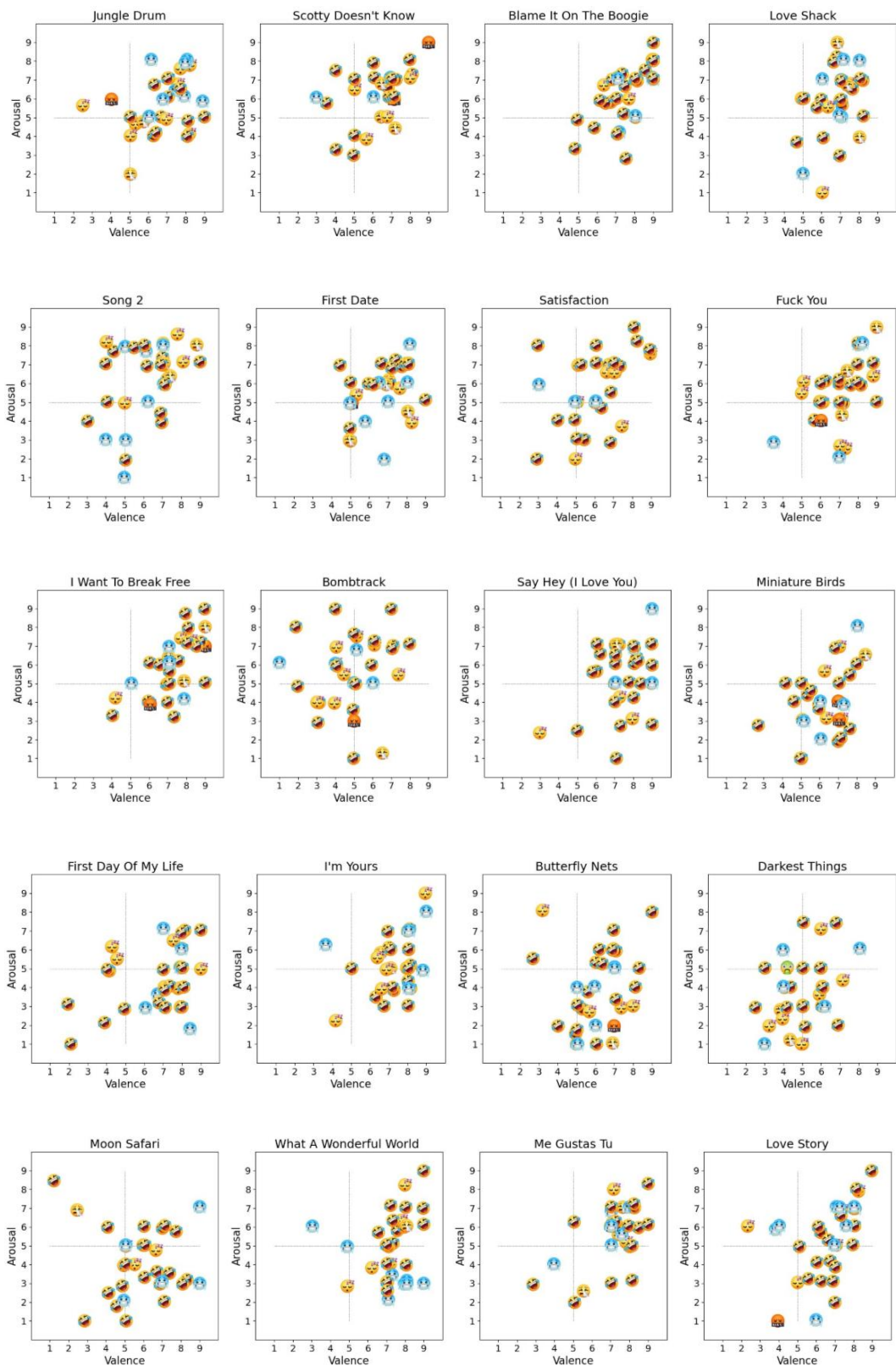


Figura 28. Piano di Russell



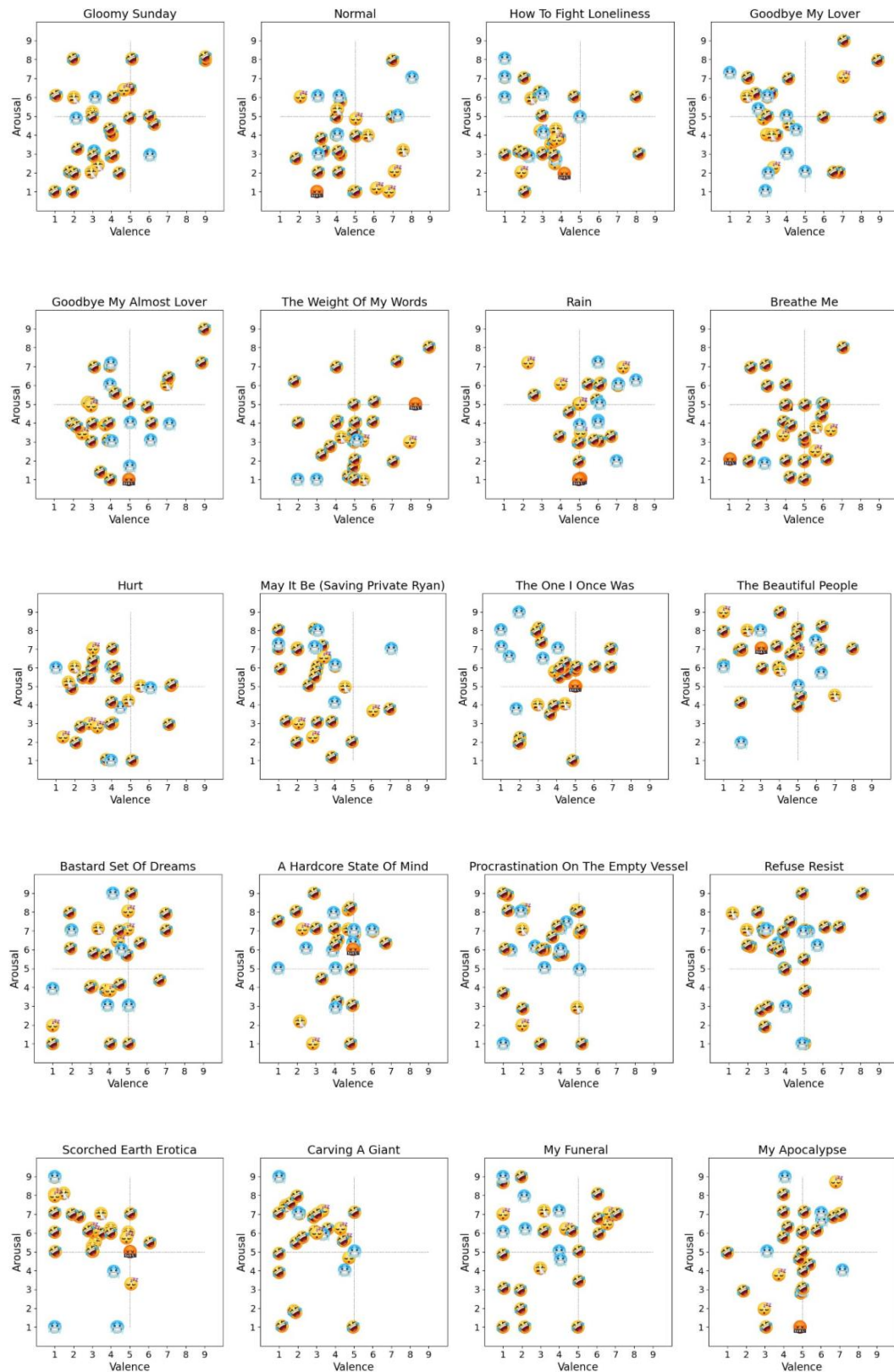
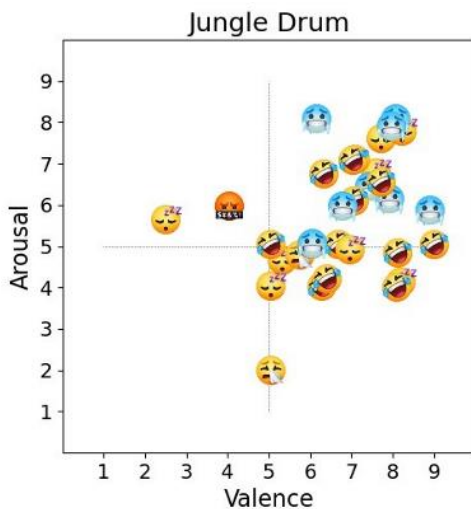


Figura 29. Grafici ottenuti dai dati DEAP

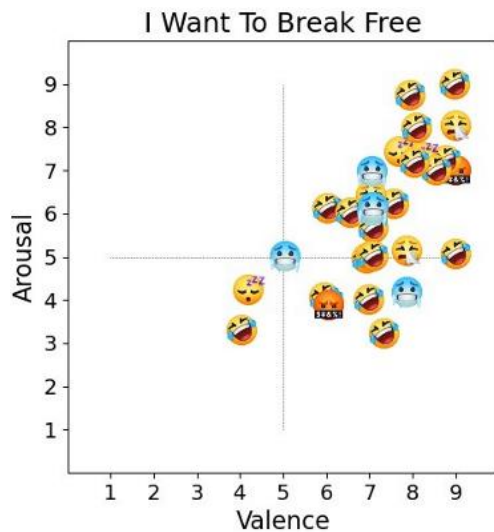
4.6. Analisi approfondita dei grafici

Osservando il piano di Russell in figura 28 e i grafici appena visti (figura 29) è evidente che molte delle emoji si trovino in posizioni errate, infatti verso il centro non ci dovrebbe essere nulla, tuttavia i quadranti individuati dalle coordinate di Valence e Arousal sono in genere plausibili. Di seguito si è posta l'attenzione su alcune delle tracce musicali più famose ignorando le emoji intorno all'origine. Da notare che il relax, il benessere, calma, felicità ed entusiasmo sono tutti indentificate con l'emoji della felicità dal Classificatore considerandoli come vari livelli della stessa.

Jungle Drum



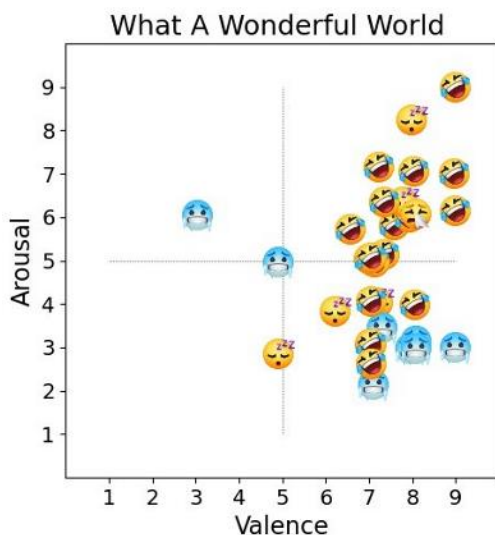
I Want To Break Free



tutte le emozioni di felicità (una è stata collocato in posizione errata), per quanto riguarda l'ansia e la tristezza è più probabile che sia un errore di previsione mentre per le restanti non è possibile fare ipotesi.

Famosissima canzone pop rock dei Queen pubblicata nel 1984, è stata la prima nella classifica europea e sudafricana. Per la posizione nel grafico sembra che i partecipanti abbiano provato felicità ed entusiasmo (primo quadrante), alcuni serenità e relax (quarto quadrante), due partecipanti si trovano invece nell'area della noia. Anche in questo caso il Classificatore ha previsto correttamente

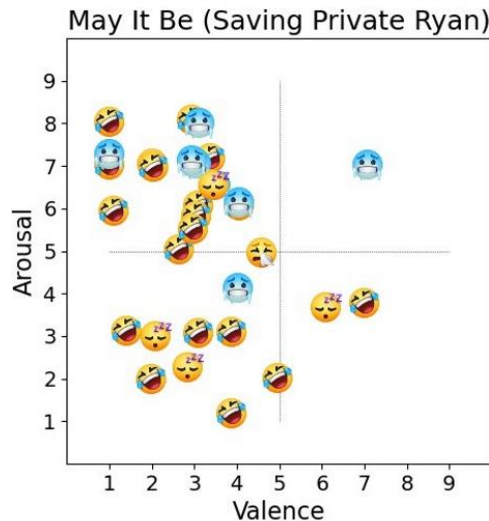
What a Wonderful World



mentre è di nuovo probabilmente in errore per le emoji dell'ansia sul quarto quadrante considerando la canzone. Le due emoji della noia in fondo sono vicine all'area corretta (parte bassa del terzo quadrante) quindi potrebbero avere coordinate errate.

Famosa canzone jazz interpretata da Louis Armstrong e pubblicata nel 1967. I partecipanti hanno provato per la maggior parte gioia (primo quadrante), calma e relax (quarto quadrante), i tre partecipanti più a sinistra sono stati classificati correttamente anche se quello al centro è ovviamente mal posizionato. Anche questa volta il Classificatore si trova in accordo con le coordinate delle emoji della felicità,

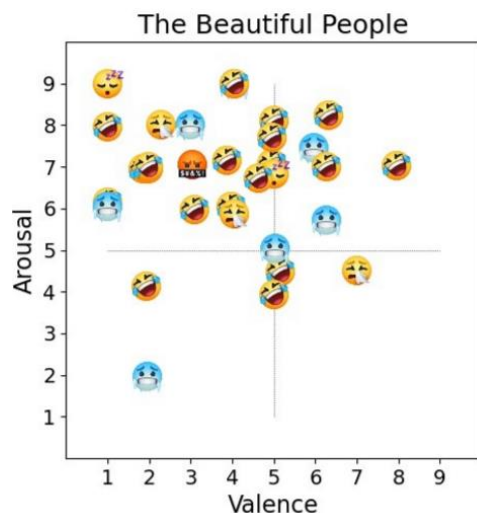
May It Be (Saving Private Ryan)



Canzone New Age del 2001 di Enya con immagini e video del film ‘Saving Private Ryan’. Secondo le coordinate la maggior parte dei partecipanti ha provato tensione, ansia, frustrazione e stress (secondo quadrante), noia e tristezza (terzo quadrante). In questo caso il classificatore ha sbagliato le emoji della felicità considerando le forti immagini di guerra e dolore presenti nel video. È però in accordo con l’emozione di

felicità (calma) del quarto quadrante mentre quella della noia accanto è probabilmente mal collocata considerando la sua vicinanza all’area corretta, stessa cosa vale per l’emoji della tristezza. Il Classificatore ha previsto correttamente la maggior parte delle emoji di ansia e di noia.

The Beautiful People



Canzone di Marilyn Manson di genere metal pubblicata nel 1996. Secondo i valori di Valence e Arousal i partecipanti hanno provato disgusto, stress, rabbia, ansia e paura (secondo quadrante), mentre alcuni l’anno apprezzata (primo quadrante). Considerato il tipo di canzone e il suo video è probabile che le emoji sul terzo e quarto quadrante abbiano tutte coordinate sbagliate, le emoji della

felicità su e vicino il primo quadrante sono classificate correttamente, così anche l’ansia, la tristezza e la rabbia del secondo quadrante. Sulle emoji di felicità e ansia non in accordo non è possibile fare ipotesi.

5. CONCLUSIONI FINALI

I video in cui il Classificatore ha ottenuto ottimi risultati prevedendo correttamente più della metà delle emoji sono ‘Blame It On The Boogie’, ‘Love Shack’, ‘Say Hey’, ‘What a Wonderful World’ e ‘Love Story’, mentre i video in cui ha ottenuto bassi risultati sono ‘The Weight Of My Words’ e ‘Corving A Giant’. Si nota facilmente come l’albero sia più efficiente nel riconoscere le emozioni positive. In generale i risultati ottenuti sono più che soddisfacenti considerando che il numero di emoji in accordo con i valori di Valence e Arousal sono oltre il 30% del totale (stima ottenuta tramite il conteggio delle emoji), inoltre essendoci moltissime emoji troppo vicine al centro del grafico (non valutabili) e per la presenza di punti in coordinate probabilmente scorrette in relazione al tipo di video, sicuramente la percentuale di accuratezza è superiore. Va anche tenuto conto dell’approssimazione che è stata fatta nell’ottenere le EEG per la mancanza dei canali richiesti dal Classificatore. Per finire è bene notare come il sistema di autovalutazione con Valence e Arousal abbia causato troppi valori ingiudicabili a causa della vicinanza al centro, che secondo il piano di Russel dovrebbe essere vuoto, per questo motivo i video ‘Rain’, ‘Brethe Me’, ‘The One I Once Was’ e ‘My Apocalypse’ sono stati scartati per le valutazioni finali. Questo suggerisce che le scale discrete utilizzate da DEAP non sono state sufficientemente comprese dai partecipanti rendendo quindi le loro autovalutazioni poco attendibili, perciò, a seguito dell’analisi appena fatta sui grafici, si può concludere che un sistema di addestramento che usa come feedback le emoji è sicuramente migliore delle scale discrete. Infatti il sistema delle emoji non può produrre valori ingiudicabili e può essere compreso immediatamente da chiunque.

BIBLIOGRAFIA

- [1] F. Q. , “Vanessa Zappalà, parla l’amica della 26enne uccisa dall’ex fidanzato stalker: ‘Le dicevo di stare attenta, avevo paura per lei’”, Il Fatto Quotidiano, www.ilfattoquotidiano.it, 2021
- [2] “Il numero delle vittime e le forme della violenza”, Istat, www.istat.it
- [3] “Omicidi di donne”, Istat, www.istat.it
- [4] “Suicidi ogni anno 200 giovani si tolgono la vita: bullismo e fallimento scolastico le cause principali”, TGCOM24, 2020
- [5] Banca Dati Statistica, INAIL, inail.it
- [6] Simplilearn, “Machine learnig Basics | What is Machine Learning? | Introduction To Machine Learning | Simplilearn”, youtube, 2018
- [7] A. Pant, “Introduction to Machine Learning for Beginners”, towards data science, towardsdatascience.com, 2019
- [8] G. Edwards, “Machine Learning | An Introduction”, towards data science, towardsdatascience.com, 2018
- [9] R. Allen, “A Gentle Introduction to Machine Learning Concepts”, machine learning in practice, medium.com, 2020
- [10] V. Zhou, “Machine Learning for Beginners: An Introduction to Neural Networks”, towards data science, towardsdatascience.com, 2019
- [11] A. Asperti, “An introduction to Neural Networks and Deep Learning”, “Talk given at the Department of Mathematics of the University of Bologna, 2018
- [12] M. Nielsen, “CHAPTER 1: Using neural nets to recognize handwritten digits”, neuralnetworksanddeeplearning.com, 2019

- [13] C. Woodford, “Neural networks”, EXPLAINTHATSTUFF!, www.explainthatsruff.com, 2021
- [14] codebasics, “Machine Learning Tutorial Python – 9 Decision Tree”, youtube, 2018
- [15] M. Galarnyk, “Understanding Decision Trees for Classification (Python)”, towards data science, towardsdatascience.com, 2019
- [16] G. Seif, “A Guide to Decision Trees for Machine Learning and Data Science”
- [17] muse, choosemuse.com/muse-s
- [18] “Electroencephalography”, www.bem.fi/book/13/13.htm, fig. 13.2
- [19] shimmer discovery in motion, www.shimmersensing.com/products
- [20] ResearchGate, www.researchgate.net/figure/Geneva-emotion-wheel-Source-Tran-2004_fig2_327309864
- [21] DEAP dataset, www.eecs.qmul.ac.uk/mmv/datasets/deap/readme.html
- [22] S. Kolestra, C. Muehl, 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, vol. 3, no.1, pp- 18-31, 2012
- [23] Jungle Drum, Wikipedia, en.wikipedia.org/wiki
- [24] I Want to Break Free, Wikipedia, it.wikipedia.org/wiki
- [25] What a Wonderful World, Wikipedia, it.wikipedia.org
- [26] May It Be, Wikipedia, it.wikipedia.org
- [27] The Beautiful People, Wikipedia, it.wikipedia.org