UNIVERSITÀ DI PISA

Dipartimento di Filologia, Letteratura e Linguistica

Corso di Laurea Magistrale in Informatica Umanistica

Predittori di attenzione e di categorie tematiche: uno studio sulla comunicazione in un corpus multimodale di visite turistiche guidate

Candidata
Ludovica Binetti

Relatore
Prof. Felice Dell'Orletta

Correlatore
Dott. Andrea Amelio Ravelli

Controrelatore
Prof. Mirko L. A. Tavosanis

Anno accademico 2020-2021

«Love is not love
Which alters when it alteration finds,
Or bends with the remover to remove.
O no! It is an ever-fixed mark
That looks on tempests and is never shaken»
(Shakespeare, Sonetto 116)

A te, Jolyon,
che non hai mai smesso di credere in me.

Ringraziamenti

Non avrei potuto trovare delle persone che mi avrebbero seguita e guidata come hanno fatto, settimana dopo settimana, il professore Dell'Orletta e il dottore Ravelli in questo "quasi anno" trascorso insieme. A loro, dunque, va un enorme grazie per avermi supportato in questi lunghi mesi di studio e lavoro, incoraggiandomi e fornendomi preziosi consigli e suggerimenti.

Durante i miei studi ho avuto anche la fortuna di immergermi in un clima stimolante e collaborativo fatto di moltissimi studenti appassionati con i quali ho condiviso uno straordinario percorso di crescita accademico e personale. Nonostante la situazione medico-sanitaria mi abbia impedito di vivere appieno gli ultimi anni da studentessa universitaria, con molti di loro ho costruito un rapporto di amicizia destinato a durare nel tempo. Ringrazio, quindi, i miei principali compagni di viaggio: Angelica, Lorenzo, Stefano e Lucia, perché senza di loro non sarebbe stato lo stesso.

In questa sede sento di dover ringraziare anche la mia famiglia. In particolare, i miei nonni, i miei primi maestri di vita, che mi hanno insegnato che la felicità risiede nelle piccole, semplici e umili cose. Mia zia Eva, la mia *partner in crime*, sempre pronta a darmi la giusta carica per proseguire dritta lungo la mia strada. Mio fratello Nicola che, orgoglioso dei miei successi, mi segue da lontano, costantemente, con l'occhio vigile che solo un fratello può avere. Mio padre, la persona più saggia e coraggiosa che io conosca, che mi insegna, con insieme durezza e dolcezza, ad affrontare le difficoltà della vita. Infine, mia madre.

A te, mamma, non dedico la Tesi ma la conclusione dell'*intero* mio percorso di studi: tutti i traguardi che ho raggiunto in questi 5 anni (e che so non saranno gli ultimi) li devo a *te*, la mia roccia, la mia spalla, il mio rifugio sicuro... La sola persona che riesca a comprendere le mie fragilità e a dare, a queste, un senso. Non cerchi di cambiarmi ma mi accetti per quella che sono. Mi hai lasciato andare a inseguire il mio futuro, i miei sogni, continuando, però, a rimanermi a fianco... Di questo ti sarò, per sempre, infinitamente grata.

Indice

Capitolo 1

Introduzione

Da sempre si è cercato di capire se la comunicazione fosse una capacità interamente umana o propria di altri esseri viventi, spesso paragonando le nostre abilità linguistiche e comunicative con quelle dei primati più simili a noi.[1] Se per lungo tempo si è dibattuto su questi argomenti, negli ultimi anni si è assistito alla diffusione capillare di sistemi artificiali in grado di elaborare il linguaggio naturale quasi diffondendo l'idea per cui esso non fosse più una peculiarità solo umana.

Al giorno d'oggi siamo, infatti, circondati da sistemi intelligenti in grado di comprendere delle richieste, di soddisfarle oppure di porre delle domande: si pensi ad esempio ad assistenti virtuali come *Google Home* o *Alexa* oppure ai sempre più diffusi *chatbot* che vengono impiegati in molti settori per raccogliere dati sulla qualità di un servizio intervistando telefonicamente i clienti.

Grazie al successo che tecnologie interattive e virtuali stanno avendo nel settore ludico, dell'educazione e dei beni culturali, l'utilizzo di simili agenti si sta ampliando fino a raggiungere contesti più propriamente educativi e divulgativi. Ciò è dimostrato dal sorgere di progetti come CHROME che punta alla realizzazione di agenti artificiali dotati di strategie comunicative (verbali e gestuali) equiparabili a quelle di una guida museale esperta.

Del progetto CHROME, delle sue caratteristiche e finalità nonché degli enti coinvolti e della tipologia di materiali raccolti si parlerà nel capitolo 2.

Da questo ricchissimo e articolato progetto sono nati tutta una serie di micro-interessi e di ricerche parallele volte a indagare diversi aspetti della comunicazione a partire dai dati raccolti in seno a CHROME. Nel capitolo 3 si parlerà di alcuni di essi relativi, in particolare, al fenomeno di attenzione a cui si è interessato l'Istituto di Linguistica computazionale "Antonio Zampolli" di Pisa.

Dopo aver fornito quasi una sorta di "background teorico" entro cui il presente lavoro di Tesi ha preso forma, si procederà a definire il vero oggetto di interesse di questo studio che mira, in generale, ad individuare gli aspetti lingui-

[1] Uno dei casi più famosi è quello di Kanzi: il primo bonobo della storia ad aver sviluppato, a seguito degli addestramenti della studiosa Sue Savage-Rumbaugh, un sistema di comunicazione basato su lessicogrammi. Cfr. Savage-Rumbaugh e Lewin (1994).

stici e fonetico-acustici più salienti della comunicazione attraverso l'annotazione di un corpus multimodale e l'utilizzo di quest'ultimo per l'addestramento di due classificatori: uno di attenzione e uno di categorie tematiche.

Nel capitolo 4 verranno descritte in specifico tutte le fasi che hanno portato alla messa in piedi del corpus in questione a partire da alcune registrazioni audio derivanti dai dati CHROME. Una delle questioni più spinose è stata la definizione delle unità frasali che avrebbero composto il dataset: individuare, a partire da un discorso parlato, un segmento eleggibile allo status di frase non è, infatti, un task banale, ma, al contrario, un'operazione che implica, oltre la conoscenza nativa della lingua, anche una serie di criteri da seguire e prendere in considerazione. Una volta conclusa questa delicata fase di *speech segmentation* è stato possibile procedere all'annotazione di ogni frase del corpus con informazioni linguistiche, fonetico-acustiche e categoriali secondo modalità specifiche che verranno accuratamente descritte nel capitolo sopra menzionato.

Si procederà poi, nel capitolo 5, ad introdurre tutta una serie di concetti basilari derivanti dal *Machine Learning* e utili alla comprensione degli aspetti architetturali di sistemi automatici per la classificazione.

Nei capitoli successivi (6 e 7), verranno, infatti, descritte le fasi di costruzione, addestramento e valutazione delle performance dei due classificatori di interesse: come prima dichiarato, da un lato il classificatore di attenzione (fenomeno descritto nel paragrafo 3.1) e dall'altro quello di categorie tematiche (volto alla classificazione del tema argomentale di una frase sulla base delle sue caratteristiche sintattico-acustiche).

Nella Conclusione (capitolo 8) si cercherà, infine, non solo di evidenziare i limiti e i possibili miglioramenti applicabili al presente lavoro ma anche di mettere insieme i risultati dei due esperimenti condotti al fine di fornire una riflessione sul concetto di "comunicazione efficace" e individuare eventuali aspetti del discorso in grado di definirla.

Prima di concludere questa Introduzione, si vorrebbe tuttavia riflettere sulle possibili applicazioni pratiche di uno studio volto a indagare gli aspetti di cui sopra si è parlato.

Non c'è dubbio che l'attuale indagine abbia preso forma da un settore estremamente specifico: quello dei beni culturali. Eppure, la speranza è che i risultati possano rivelarsi utili per altre tipologie di ambiti. L'ambito educativo sarebbe, ad esempio, uno dei tanti a beneficiarne maggiormente: ascoltare una persona (un professore) in grado di mantenere alto il livello di attenzione di coloro che ascoltano (gli studenti) e in grado di veicolare in maniera efficace i contenuti agevolerebbe, senz'altro, il processo di apprendimento.

Il "saper comunicare" è, però, una prerogativa indispensabile in molti altri settori. Tecniche di comunicazione efficaci potrebbero essere d'aiuto non solo a docenti, ma anche a guide turistiche, relatori di congressi e così via. A quanti, ad esempio, è capitato di prendere parte a una conferenza e convenire che il relatore, per il suo ritmo troppo lento, per il suo tono troppo basso o per la sua cadenza eccessivamente monotona, non era in grado di mantenere viva l'attenzione della sua audience o di "farsi capire"?

Oltre ad "addestrare" le persone all'arte di "saper comunicare", una tale indagine potrebbe fornire un contributo significativo nella definizione e messa in piedi di sistemi artificiali in grado di elaborare il linguaggio naturale, argomento sul quale ci si vorrebbe qui soffermare.

Come accennato all'inizio di questa Introduzione, se è vero che, da un lato, i sistemi dotati di competenze linguistiche sono sempre più diffusi e parte integrante della vita quotidiana, dall'altro non è scontato che un fruitore umano abbia un'esperienza positiva nel loro utilizzo.

Un'idea che si ritrova in Sidner et al. (2005) è che, quando si interagisce, i protagonisti di un'interazione modificano il loro comportamento, adattando la loro strategia comunicativa sulla base delle reazioni di coloro che ascoltano, al fine di "non lasciarli andare via", ovvero al fine di generare e mantenere in loro una soglia minima di interesse e coinvolgimento.[2]

> «Il punto è che, quando le persone parlano, mantengono una volontaria e reciproca connessione psicologica tra loro e l'una non lascerà andare via l'altra» (Sidner et al. 2005, p. 141)[3]

Se gli umani sono, dunque, in grado di instaurare tra loro questa "connessione", cosa succede, invece, nelle interazioni macchina-uomo?[4]

«Le macchine», continuano Sidner e colleghi, «non fanno niente di tutto ciò», nel senso che non sono in grado di mettere in pratica quell'«insieme di regole non dette» che tutti noi umani non solo conosciamo inconsciamente, ma sappiamo anche applicare.[5]

In altri termini, le macchine non conoscono l'*etiquette* della comunicazione tra umani e questo potrebbe costituire un problema in tutti quei settori che cercano di innovarsi e promuoversi mediante un uso più sistematico delle nuove tecnologie, come appunto gli agenti conversazionali.

Nel settore dei beni culturali progetti come CHROME mirano, come accennato, alla realizzazione di guide artificiali in grado di accompagnare i visitatori in dei tour virtuali. Questi ultimi darebbero la possibilità di risolvere non solo i problemi di accessibilità da parte, ad esempio, di persone diversamente abili, ma anche di risolvere gli impedimenti connessi a un periodo difficile e particolare come quello che si è vissuto a causa della diffusione della malattia SARS-CoV-2.

[2] Questo comportamento, sostengono Sidner e colleghi, riguarda tutti i partecipanti a un discorso, sia quelli che parlano ma anche quelli che ascoltano. Tuttavia, nel caso specifico di questo *case study*, ci si focalizzerà in un tipo di comunicazione uno a molti, in cui i "molti" hanno un ruolo prevalentemente passivo, di semplici uditori. Pertanto, delle riflessioni presenti in Sidner et al. (2005), si considera solo ed esclusivamente l'aspetto per cui è responsabilità dello speaker instaurare (e mantenere) una connessione con l'ampio pubblico al quale ci si rivolge e non il contrario.

[3] «The point is that when people talk, they mantain conscientius psychological connection with each other and each will not let the other person go».

[4] La scelta di invertire l'usuale forma "interazione uomo-macchina" è dovuta a quanto detto *supra* (nota 2): nel caso specifico di questo studio ci si sta riferendo a un tipo di comunicazione unico senso per cui una macchina si rivolge a un pubblico ampio di umani. Dunque è compito della macchina mantenere vivo l'interesse di quest'ultimo.

[5] Sidner et al. (2005, p. 141): «we have this set of unspoken rules that we all know unconsciously but we all use in every interaction [...] machines do none of the above».

Agenti artificiali e ambienti virtuali si porrebbero, cioè, alla base della fruizione da remoto di contenuti culturali a condizione, però, che i primi abbiano un livello di competenza linguistica e di capacità espressive tali da interagire il più naturalmente possibile con gli utenti finali.

L'idea di dotare i sistemi artificiali di competenze e capacità linguistiche equiparabili (o comunque simili) a quelle di un umano non è nuova, ma ricorda, ad esempio, il famoso gioco dell'imitazione proposto da Alan Turing e reso pubblico nel 1950 sul giornale inglese *Mind*.[6] Per rispondere alla domanda "Le macchine possono pensare?", Turing propose l'*imitation game*, un gioco in cui un interrogante è tenuto a porre domande ai partecipanti al gioco, che hanno la possibilità di rispondere per forma scritta in maniera diretta o tramite un intermediario. Lo scopo dell'interrogante è quello di capire se la persona, dall'altro lato del canale di comunicazione, è un umano o una macchina. Per poter vincere il gioco la macchina dovrebbe essere in grado di simulare le risposte che un umano darebbe alle domande dell'interrogante.

Come si capirà nel corso della trattazione, i sistemi di intelligenza artificiale vengono programmati per eseguire task specifici e sono solitamente dotati di un set di comportamenti predefiniti. Imitare un umano nei suoi comportamenti linguistici non è, però, facile non solo perché le situazioni comunicative sono potenzialmente infinite, ma anche perché la comunicazione in sé è un fenomeno estremamente complesso fatto di parole, gesti, movimenti, espressioni e tutta un'altra serie di "indizi" che non passano inosservati agli occhi di un umano. Individuare, dunque, un modello per descrivere e formalizzare questi fenomeni (da fornire, poi, a un sistema di intelligenza artificiale) è un task più arduo di quel che possa sembrare, eppure necessario. È proprio in questa esigenza che il presente studio trova una sua possibile applicazione pratica.

L'idea che ci siano dei fattori che definiscono i contorni di una "comunicazione efficace" non è, tra le altre cose, un'idea recente, ma, al contrario, risalente alla Roma del I secolo a.C.

> «Cosa c'è di più piacevole da apprendere e da ascoltare di un discorso elegante, fondato su saggi concetti ed espressioni appropriate? [...] Il discorso si deve reggere non solo su una attenta selezione dei termini, ma anche su una precisa organizzazione delle sue parti; bisogna inoltre avere una profonda esperienza della vasta gamma di stati d'animo che appartengono per natura agli uomini, poiché la forza e l'intelligenza di un valido oratore saranno messe alla prova dalla capacità di placare o stimolare le emozioni di coloro che ascoltano. [...] Che dire poi della capacità comunicativa, che emerge dai movimenti del corpo, dai gesti, dall'espressione del volto, dal controllo e dalla modulazione della voce?» (Cicerone)[7]

Studiare questi aspetti, oggi, nel mondo delle tecnologie e di agenti artificiali dotati di capacità linguistiche, si rivela più interessante che mai, indipendentemente da quella che sarà, poi, la loro applicazione pratica.

[6] Turing (1950).
[7] Cicerone (2007, a cura di Paolo Marisch).

Certo è che i dispositivi intelligenti in grado di elaborare il linguaggio naturale non vengono attualmente sfruttati nel pieno delle loro potenzialità perché ancora molto rimane da fare per renderli dei "validi oratori". Per far questo li si dovrebbe dotare o della capacità di valutare le reazioni dei propri uditori per renderli in grado di agire conseguentemente, oppure cercare di individuare una strategia comunicativa che sia valida in qualsiasi situazione e per qualsiasi interlocutore. Se la prima opzione sembra ancora troppo lontana dal realizzarsi, la seconda potrebbe essere alle porte se studi come questo, e (si spera) altri che verranno, saranno in grado di individuare gli aspetti dell'elocuzione e della declamazione che renderebbero apprezzabile un interlocutore, artificiale o umano che sia.

Capitolo 2

Il progetto CHROME

Il progetto di tirocinio da cui ha avuto origine la presente trattazione si colloca nell'ambito di un progetto triennale, ben più ampio, svoltosi tra il 2017 e il 2020: CHROME (Cutugno et al. 2018).[1] Il *Cultural Heritage Resources Orienting Multimodal Experience* ha rappresentato un vastissimo ambito di ricerca finalizzato alla raccolta di materiali di varia natura, tutti però con una comune provenienza (il mondo dei beni culturali) e un comune destino (la realizzazione di agenti artificiali). Questo capitolo ha lo scopo di fornire una panoramica del progetto, capire di che cosa si sia occupato e da quali esigenze sia nato, nonché quello di individuarne gli enti promotori e ed esporre le modalità di raccolta dei dati che hanno condotto alla costruzione di un corpus multimodale di visite guidate.

2.1 Il duplice volto di CHROME

Per comprendere, almeno a grandi linee, gli obiettivi di questo enorme e variegato progetto di ricerca, si può partire dall'analisi del suo stesso "biglietto da visita": il logo di CHROME, visibile in Figura 2.1.

Quest'ultimo mette in evidente risalto la *O* di *Orienting*, suggerendo una prima volontà dei promotori del progetto: fare in modo che i beni culturali (*Cultural Heritage Resources*) orientino, ponendosene alla base, esperienze virtuali multimodali (*Multimodal Experience(s)*)[2] di cui siano protagonisti i visitatori dei siti

[1] Pagina web del progetto: `http://www.chrome.unina.it`.

[2] Piuttosto che accogliere solo la definizione classica per cui *multimodale* farebbe riferimento ai modi di comunicare (si veda ad esempio la definizione in *Multimodale* (2021)), in questa sede il termine viene inteso anche come sinonimo di *multisensoriale*, un'esperienza, cioè, che coinvolge più sensi. Con l'emergere delle tecnologie immersive (es. visori di realtà aumentata), è accresciuto l'interesse nel cercare di capire quale fosse il ruolo di queste ultime nel processo di apprendimento. Ma, più in generale, ci si chiede se le tecnologie siano in grado di migliorare il processo di acquisizione di conoscenza. A questa domanda cercano di rispondere numerosi studi, in particolare quelli nel campo dei videogiochi come strumenti didattici. In Vicuna (2017) si trovano numerose riflessioni volte a considerare il *videogame* come un medium efficace per l'apprendimento di contenuti in giovani discenti. Non è un caso, dunque, che lo scopo del

Figura 2.1: Logo del progetto CHROME.

culturali. Ciò significa che una delle finalità del progetto è, senz'altro, la virtualizzazione di ambienti al fine di fornire, agli eventuali visitatori, la possibilità di vivere un'esperienza immersiva all'interno degli stessi.

Tuttavia, le finalità di CHROME vanno al di là di una semplice ricostruzione tridimensionale di ambienti come, d'altronde, suggerisce quella sorta di triangolo isoscele rovesciato al di sotto della *O*. Quest'ultimo sta, infatti, a rivelare un altro aspetto importante, se non il fine ultimo dell'intero progetto di ricerca: la realizzazione di agenti intelligenti che non si limitino, però, a guidare i visitatori nell'esperienza multimodale di cui sopra si parlava, ma che siano in grado di comunicare e trasmettere efficacemente contenuti di una certa qualità.

Come si avrà modo di approfondire nel paragrafo 2.3, per realizzare una simile tipologia di agenti si è visto necessario dotare questi sistemi della capacità di imitare le strategie comunicative verbali (e non)[3] di guide turistiche umane, le quali, dopo anni di esperienza sul campo, sono in grado di comprendere i bisogni degli ascoltatori e di reagire ai loro stimoli.

Del duplice volto di CHROME, in questa sede, si lascerà da parte quello che riguarda la ricostruzione tridimensionale di ambienti per concentrarsi maggiormente su quello che ha più strettamente a che fare con la realizzazione di tecnologie interattive (agenti artificiali) per la fruizione di beni culturali.

2.2 Gli enti coinvolti

Le ampie finalità del progetto CHROME, che possono essere sintetizzate (e generalizzate) nella volontà di indagare la natura della comunicazione tra umani e individuarne le peculiarità al fine di realizzare un modello computazionale che le sintetizzi, hanno inevitabilmente condotto alla collaborazione di più enti di ricerca, ognuno dei quali ha messo a disposizione risorse e conoscenze specifiche. Quando si portano avanti progetti di una tale portata, infatti, è difficile

progetto CHROME sia quello di realizzare esperienze multimodali al fine di migliorare la fruizione dei beni culturali.

[3]Si vedrà più avanti, nel paragrafo 3.1, che anche la componente non verbale del linguaggio costituisce un aspetto imprescindibile della comunicazione tra umani.

trovare, all'interno di uno stesso team di ricercatori, tutte le conoscenze di cui si ha bisogno. Nel caso di CHROME erano indispensabili esperti come linguisti teorici, linguisti computazionali (con conoscenze nel campo della comunicazione non verbale, della prosodia e della pragmatica), *computer scientists* (con competenze nei campi dell'Intelligenza Artificiale e dell'interazione uomo-macchina), così come psicologi e architetti competenti nel campo delle ricostruzioni 3D.[4] La necessità, dunque, di includere esperti di discipline diverse, seppure, in qualche modo, complementari, ha condotto alla partecipazione di 5 gruppi di ricerca:

- Università degli Studi di Napoli "Federico II" (Urban/Eco) - che ha avuto il compito di raccogliere i materiali per la virtualizzazione degli ambienti e per la messa a punto del corpus multimodale;
- Istituto di Linguistica Computazionale "Antonio Zampolli" di Pisa (ILC) - a cui è stato affidato il compito di sviluppare un sistema per estrarre e organizzare la conoscenza linguistica a partire da un corpus di dati;
- Università degli Studi di Salerno (UniSa) a cui è stata affidata l'analisi prosodica dei testi con lo specifico obiettivo di utilizzarla per questioni relative alla sintesi vocale;
- Istituto di Scienze Applicate e Sistemi Intelligenti "Eduardo Caianiello" (ISASI) - che si è occupato del problema relativo alla realizzazione di un modello computazionale in grado di generare linguaggio naturale;
- Università degli Studi di Roma "RomaTre" (RomaTre) - che si è occupata dell'analisi delle componenti non verbali di un'interazione, ovvero di un'analisi più propriamente gestuale.

La presenza di un team eterogeneo ha reso possibile non solo una migliore suddivisione dei compiti ma ha anche garantito che, a ognuno di questi, si dedicassero figure professionali esperte e capaci di gestire eventuali problemi (a cui si accennerà nella sezione 2.4) che un corpus multimodale, come quello raccolto, avrebbe inevitabilmente comportato.

2.3 La raccolta dei materiali

Per realizzare un modello computazionale in grado di sintetizzare (simulare) il comportamento di guide turistiche umane è stato, prima di tutto, necessario analizzare tale comportamento al fine di individuare gli aspetti peculiari e salienti della comunicazione efficace di una guida turistica. Quale miglior modo, infatti, di insegnare a una macchina a comportarsi come un umano se non osservando il comportamento di quest'ultimo?

Per tale ragione, registrazioni audio-video di visite turistiche guidate costituiscono il materiale di partenza del progetto CHROME. Se ci si riflette, infatti,

[4] Si veda Cutugno et al. (2018, p. 2).

è ascoltando e osservando una persona che si apprende quel comportamento di cui poc'anzi si parlava. In questa sede, per *comportamento* si intende l'insieme delle movenze del corpo e dei gesti nonché le caratteristiche della voce (quali tono e velocità) della guida turistica.[5] L'interesse in una tale tipologia di informazione (e dunque, più in generale, l'interesse nel costruire un corpus multimodale) nasce sulla base del fatto che anche gli aspetti non verbali (per così dire "contestuali") di un'interazione tra umani contribuiscono all'efficacia di un atto comunicativo.[6]

Se è vero, però, che un'analisi approfondita dei comportamenti di una guida turistica, che detiene la conoscenza da trasmettere, è indispensabile per la realizzazione del modello computazionale, è anche vero che in simili contesti comunicativi la comunicazione tra umani non è quasi mai unidirezionale ma, al contrario, bidirezionale, in quanto coinvolge anche la platea degli ascoltatori offrendo loro una possibilità di confronto con la guida. Per poter analizzare a tutto tondo il fenomeno di interazione tra umani, si è visto necessario acquisire video da due diverse inquadrature, ognuna delle quali corrisponde ai due estremi di un ipotetico canale di comunicazione: il mittente, nel nostro caso la guida, e il destinatario, ovvero il pubblico. Ciò ha permesso di analizzare il comportamento di tutti i partecipanti allo specifico evento comunicativo e di studiarne, in particolare, gesti, movimenti e reazioni. Si ribadisce ancora una volta, che la peculiarità del progetto è proprio quella di analizzare il fenomeno della comunicazione nella sua complessità, non limitandosi all'analisi del solo flusso di parole ma estendendo l'attenzione anche a tutte quelle componenti cosiddette *paralinguistiche*.[7]

Da ricordare, inoltre, che il materiale audiovisivo non è l'unica tipologia di materiale raccolto. Come già accennato, sono stati collezionati anche rilievi tridimensionali degli ambienti e materiali testuali descrittivi degli stessi.

Questi ultimi hanno posto le fondamenta per la delineazione del dominio di conoscenza della guida artificiale: si tratta, infatti, di documenti di storia dell'arte, testi scientifici, cataloghi specializzati, testi provenienti da siti web certificati che forniscono la conoscenza di cui il sistema ha bisogno per generare i contenuti da trasmettere.

Da qui in poi si menzionerà solo ed esclusivamente il materiale audiovisivo in quanto, come si avrà modo di approfondire nel capitolo 4, è proprio di tale tipologia di dato che si è occupato il presente lavoro.

[5] Questa definizione di comportamento rientrerebbe in quello che Origlia et al. (2019, p. 395), in uno studio volto all'identificazione dei gesti associabili alle pause del discorso di una guida turistica, definiscono *communicative behavior*: «speech is accompanied by communicative behavior in other modalities, to the extent that the whole body is involved» («il discorso è accompagnato da altre modalità di *communicative behaviour*, a tal punto che il corpo stesso ne è coinvolto»).

[6] Come già accennato in nota 3 (par. 2.1), dell'importanza degli aspetti non strettamente verbali del linguaggio si discorrerà più avanti.

[7] La paralinguistica è una disciplina che si occupa della comunicazione non verbale (come mimica, gesti e movimenti) ma anche dei tratti soprasegmentali della lingua (ad esempio altezza, volume e tono del discorso).

2.4 L'uso di un corpus multimodale: sfide e problemi

Una volta raccolto il materiale di partenza, si è opportunamente proceduto alla sua annotazione. Come si apprende da Cutugno et al. (2018), l'avere a che fare con un corpus multimodale ha creato non pochi problemi per quel che riguarda la trasformazione di dati in conoscenza analizzabile e sfruttabile da sistemi automatici di intelligenza artificiale.[8] Tali questioni sono state affrontate, come accennato prima, grazie al dialogo e alla partecipazione degli enti sopra menzionati.

Per evitare che si vada oltre gli scopi prefissati da questa Tesi, non si andrà qui, in specifico, a parlare di quali siano state le modalità adottate per annotare il corpus.[9] Tuttavia, per avere almeno un'idea dei problemi affrontati si pensi a quanto sia complesso, a partire da registrazioni audio-video, trascrivere fedelmente la lingua parlata;[10] estrarre da questa contenuti che possano essere, in una fase successiva, utilizzati per la generazione di testo da parte di sistemi automatici per la sintesi vocale; oppure effettuare un'analisi prosodica dei segmenti audio per dotare i suddetti sistemi di una competenza paralinguistica simile a quella di un umano; o ancora analizzare i video annotando gesti e movimenti che la guida turistica artificiale dovrebbe essere in grado di simulare.

Un'ulteriore grande sfida è stata, inoltre, analizzare il comportamento "adattivo" della guida turistica umana per comprendere le diverse tipologie di strategie comunicative messe in atto durante la presentazione di contenuti culturali a diversi gruppi di visitatori.

Le questioni di cui si discorrerà più in dettaglio nei capitoli 3 e 4 sono quelle affrontate dall'Istituto di Linguistica Computazionale di Pisa (come si vedrà, in un ambito leggermene diverso dalle originarie finalità di CHROME) e relative, in particolare, all'annotazione dell'*attention* e all'individuazione di segmenti frasali a partire dalla trascrizione di lingua parlata.

2.5 Il caso studio

Con in mente gli obiettivi di cui sopra si è discusso, è stato, ovviamente, necessario individuare un *case study* che consentisse di raccogliere, il più agevolmente possibile, materiali per la costruzione di un corpus sul quale, poi, lavorare per la realizzazione di quello che in Cutugno et al. (2018) viene definito un *Gatekeeper Computational Model*.

La scelta è ricaduta su 3 diverse Certose presenti all'interno del territorio campano: la Certosa di San Martino a Napoli, la Certosa di San Lorenzo a

[8] Nel paper citato nel testo si parla delle difficoltà di convertire i materiali testuali in *knowledge resources*.

[9] Per un maggiore approfondimento si rimanda a Cutugno et al. (2018).

[10] Nonostante la trascrizione avvenga, oggigiorno, tramite sistemi di trascrizione automatica, le performance di questi strumenti non sono ancora del tutto perfette e necessitano di una revisione da parte di un annotatore umano.

Padula e la Certosa di San Giacomo a Capri. Presso queste meravigliose strutture si sono raccolti i materiali di partenza di cui si è parlato nel paragrafo 2.3. Nonostante i dati siano stati collezionati per ogni Certosa, di fatto la maggior parte degli sforzi dei vari gruppi di ricerca si sono concentrati prevalentemente sulle registrazioni audio-video della sola Certosa di San Martino di Napoli. Per tale ragione, da qui in avanti, ci si riferirà solo ed esclusivamente a quest'ultimo sottoinsieme di materiali.

Figura 2.2: L'immagine (presa da *Certosa di San Martino* (2021) e opportunamente modificata) mostra la piantina della Certosa di San Martino di Napoli. In essa è possibile identificare i 6 ambienti principali: 1) Pronao; 2) Chiostro grande; 3) Parlatorio; 4) Sala del Capitolo; 5) Coro; 6) Stanza del Tesoro.

Nello specifico, sono state registrate 12 visite, guidate rispettivamente da 3 diverse guide turistiche di genere femminile.[11] Alle visite hanno preso parte gruppi di 4 visitatori che si è cercato di selezionare sulla base di genere e provenienza socio-demografica.[12] Inoltre, il percorso guidato presso la Certosa di San

[11] La scelta di non raccogliere materiali di guide di entrambi i generi è giustificata dalla necessità di mantenere l'analisi il più semplice possibile. Avere guide di genere maschile e femminile avrebbe significato dover tenere conto di tutta una serie di variabili specifiche di genere (es. range di frequenze del tono della voce) che sarebbe stato troppo complesso gestire.

[12] L'obiettivo di una tale selezione era quello di rendere i vari gruppi il più possibile bilanciati in merito al primo aspetto e il più possibile variegati in merito al secondo.

Martino prevede l'attraversamento di 6 ambienti principali, che costituiscono dei *punti di interesse* (POI da *Point Of Interest*), ovvero dei punti in cui la guida si sofferma a parlare fornendo ai visitatori informazioni circa l'ambiente circostante.

Questi ambienti, visibili in Figura 2.2, sono:

1. Il Pronao: luogo antistante l'ingresso della chiesa in cui la guida offre un'introduzione alla Certosa e al museo;
2. Il Chiostro: ampio spazio esterno vicino al cimitero dei monaci. Vengono fornite ai visitatori informazioni relative alla vita dei certosini;
3. Il Parlatorio: il primo ambiente interno previsto dalla visita. La guida, oltre a fornire ulteriori informazioni circa lo svolgimento della normale vita in Certosa, descrive gli affreschi che abbelliscono questo ambiente;
4. La Sala del Capitolo: stanza adiacente al Parlatorio. La guida continua nell'esposizione delle rigide regole caratterizzanti la vita dei monaci e nella descrizione degli elementi artistici e architettonici presenti nella stanza;
5. Il Coro: posizionato dentro la chiesa, alle spalle dell'altare. La guida illustra gli elementi artistici e architettonici caratterizzanti questo ambiente;
6. La Stanza del Tesoro: ambiente conclusivo la visita guidata.

Figura 2.3: I dati CHROME su cui si è maggiormente lavorato.

Del materiale audiovisivo raccolto si è condotta, prevalentemente, un'analisi del parlato incentrata su diversi livelli: livello ortografico, fonetico, sillabico, intonativo, testuale e, infine, multimodale (livello nel quale rientra l'annotazione di gesti, movimenti del corpo ed espressioni facciali).[13] Quest'ultima è stata effettuata mediante un opportuno software per l'annotazione multimodale: ELAN, di cui si tratterà più dettagliatamente nel paragrafo 4.2.1.1.

[13]Per maggiori approfondimenti sulle caratteristiche di ogni singolo livello di analisi si rimanda a Cutugno et al. (2018).

Per una visione schematica della tipologia di dati usati in seno a CHROME si rimanda alla Figura 2.3.

2.6 Lo stato dell'arte

L'articolo di Cutugno et al. (2018), che è stato più volte citato nel corso di questo capitolo, fornisce lo status dei lavori alla data di luglio 2018. Dopo la loro conclusione, avvenuta nel 2020, alcuni dei risultati sono stati pubblicati in un sito dedicato[14] che, tutt'oggi, continua ad essere aggiornato.

Qui si possono trovare diverse, stimolanti, pubblicazioni che indagano una varietà di aspetti e fenomeni collegati ai temi di cui CHROME si è occupato negli anni. Tra i tanti studi, si annoverano quelli che indagano sui possibili usi delle tecnologie nel campo dei beni culturali con il dichiarato scopo di migliorare l'esperienza dei visitatori in un museo;[15] oppure quelli interessati a indagare le caratteristiche fonetiche e prosodiche del parlato eventualmente correlate a funzioni pragmatiche;[16] o ancora, quelli il cui obiettivo è esaminare tutti gli aspetti, linguistici, fonetici, gestuali e, più in generale, comportamentali,[17] in grado di rendere la guida artificiale il più umana possibile.[18] Ovviamente, accade spesso che questi temi si sovrappongano e che li si ritrovi anche all'interno di una stessa pubblicazione.

Nella sezione *Multimedia* del sito è possibile, inoltre, visionare:

- Alcuni prodotti relativi ai modelli tridimensionali realizzati (in particolare la vista dei complessi delle tre Certose);
- Un video che offre una panoramica dello strumento usato per l'annotazione multimodale (ELAN);
- Un video demo dell'avatar sviluppato.

Infine, a breve, sarà discussa una Tesi di dottorato, a cura di Loredana Schettino, che dovrebbe fornire una panoramica completa e definitiva di quello che è stato il progetto CHROME negli anni tra il 2017 e il 2020.

[14] Riferimento web: `http://www.chrome.unina.it/`.
[15] Ad esempio, si consulti Cera (2020) oppure Sorgente et al. (2017).
[16] Si segnalano Schettino e Cataldo (2019) e Ansani (2019).
[17] Cfr. *supra*, par. 2.3.
[18] Si rimanda a Origlia et al. (2019) e Cataldo et al. (2019).

Capitolo 3

Un progetto derivante da CHROME

Il corpus multimodale prodotto a conclusione del progetto CHROME ha fornito una fonte di dati considerevole, che ha inevitabilmente aperto le porte a ulteriori interessanti indagini: tra queste vi è uno studio, promosso e avviato dall'Istituto di Linguistica Computazionale "Antonio Zampolli" (ILC-CNR) di Pisa, relativo all'*engagement*. L'interesse in questo fenomeno non nasce, dunque, propriamente nell'ambito di CHROME, ma ne costituisce la naturale prosecuzione, dimostrando la ricchezza di spunti e riflessioni che il corpus multimodale ha fornito anche negli anni successivi alla chiusura ufficiale del progetto.[1]

In questo capitolo, si fornirà una breve panoramica della letteratura di riferimento circa il fenomeno dell'*engagement*, nonché di alcuni lavori sull'*engagement* precedentemenți avviati e (parzialmente) conclusi in seno all'ILC.

3.1 Cosa si intende per *engagement*

Quando si parla di comunicazione, si fa riferimento all'atto di trasmettere delle informazioni di qualsivoglia natura. Se si provasse a osservare e ad analizzare il comportamento di due o più persone che comunicano tra loro, ci si renderebbe conto, quasi istantaneamente, che la comunicazione è un fenomeno estremamente complesso che non interessa esclusivamente il canale di comunicazione primario e prediletto dagli umani (ovvero il canale verbale), ma, al contrario, in esso entrano in gioco numerosi altri fattori.[2]

[1] Per una panoramica di altri studi condotti sui dati CHROME (e non solo) si rimanda *supra*, par. 2.6. Per le singole pubblicazioni si consulti la pagina ufficiale del progetto: `http://www.chrome.unina.it/publications/`.

[2] La conversazione altro non è che l'evento "prototipico" dell'atto comunicativo, ma ciò non toglie che ve ne possano essere degli altri. Il linguaggio non verbale, di cui pure gli esseri umani sono dotati, è oggetto di studio di numerose discipline tra cui la prossemica, la cinesica e la paralinguistica. La prossemica si occupa di indagare il valore comunicativo della disposizione dei corpi nello spazio. La cinesica indaga la mimica e la gestualità come

Durante un'interazione *face-to-face* (ma, riflettendoci, anche durante un'interazione a distanza) comprendere un messaggio veicolato attraverso il canale orale implica molto di più che la semplice abilità di "decifrare una stringa di caratteri".[3] Le informazioni linguistiche veicolate costituiscono, cioè, solo la minuscola punta di un iceberg[4] (qual è la comunicazione), che nasconde, al di sotto, una grande molteplicità di fattori che la influenzano: gesti, espressioni facciali, sguardo, movimenti del corpo, etc. Tutti questi sono importanti indicatori[5] di quanto sia interessata e/o coinvolta la persona con la quale si sta comunicando. Avere una simile informazione circa il proprio interlocutore è importante, in quanto il soggetto primario dell'interazione, ovvero colui che trasmette il messaggio, potrebbe decidere di modificare, o meglio adattare, la propria strategia comunicativa: ad esempio, potrebbe decidere di aggiungere più informazioni, se l'interlocutore si mostra interessato, oppure ometterle nel caso contrario.

L'interesse o il coinvolgimento manifestato da una persona nel corso di un'interazione prende il nome di *engagement*. In realtà, definire quello che, a tutti gli effetti, è un «processo fondamentale che sottosta ogni interazione umana» (Rich et al. 2010, p. 375)[6], è molto più complesso di quanto, all'apparenza, possa sembrare. Proprio per il fatto che, a detta di Rich e colleghi, l'*engagement* è un fenomeno che caratterizza qualsiasi tipo di interazione umana, esso assume delle sfaccettature e, di conseguenza, delle definizioni sempre diverse a seconda del particolare contesto in cui ci si trova. Uno dei campi di ricerca maggiormente interessato allo studio dell'*engagement* è il settore dell'educazione scolastica. Fredricks, Blumenfeld e Paris (2004), ad esempio, identificano 3 diverse tipologie di *engagement* per ognuna delle quali forniscono specifici esempi presi dal mondo della scuola: *engagement* comportamentale, emozionale e cognitivo. Similmente definiscono l'*engagement* Goldberg et al. (2019) in uno studio volto a individuare gli indicatori di questo fenomeno nei ragazzi delle scuole. Anche Goldberg e colleghi fanno una distinzione tra *cognitive*, *emotional* e *behavioral engagement*, intendendo con il primo un meccanismo cognitivo messo in atto dall'apprendente per filtrare e selezionare le informazioni provenienti dall'esterno; con il secondo una serie di aspetti che più hanno a che fare con le emozioni (e dunque non necessariamente percepibili, come noia o curiosità); e con l'ultimo tutta una serie di comportamenti osservabili dall'esterno (ad esempio partecipazione attiva alla conversazione, costanza nell'ascolto, etc).

Altro settore di ricerca interessato a indagare questo complesso fenomeno è

parti integranti della comunicazione. La paralinguistica si occupa dell'analisi del potenziale comunicativo della voce.

[3] Cfr. Ravelli, Origlia e Dell'Orletta (2020, p. 1): «understanding a message expressed through the speech channel in face-to-face interactions involves more than the ability to decipher a string of characters and to assign a meaning to words and sentences» («comprendere un messaggio espresso attraverso il canale orale durante un'interazione "faccia a faccia" implica molto di più che l'abilità di decifrare una stringa di caratteri e assegnare un significato alle parole e alle frasi»).

[4] Cfr. *ibidem*: «the linguistic information conveyed by lexicon is only the the tip of the iceberg» («l'informazione linguistica veicolata dal lessico è solo la punta di un iceberg»).

[5] Non a caso, più volte, definiti da Goldberg et al. (2019) dei «cues» («indizi»).

[6] «Fundamental process that underlies all human interaction».

il campo che si dedica allo studio delle interazioni uomo-macchina. In questi casi, l'*engagement* viene spesso genericamente definito come quel «processo attraverso cui due individui iniziano, mantengono e terminano una connessione tra loro» (Sidner et al. 2005, p. 141).[7] Inoltre, un interessante studio di Rich et al. (2010), sempre nel settore della robotica, ha condotto all'identificazione di 4 tipi di eventi (da loro definiti *connection events*) che starebbero alla base del mantenimento dell'*engagement* durante una conversazione:

- *Directed gaze*, sguardo diretto - interazione per cui una persona guarda un oggetto, eventualmente indicandolo, e l'interlocutore volge su di esso il proprio sguardo;
- *Mutual facial gaze*, scambio reciproco di sguardi - la persona che interloquisce guarda direttamente il suo interlocutore, che, a sua volta, sostiene lo sguardo. Si tratta di quello che in inglese viene definito *eye contact*;
- *Adjaceny pairs*, coppie contigue - interazioni che richiedono la collaborazione di ambo le parti. Un esempio sono domande che richiedono una risposta, spesso anche breve, o che necessitano anche solo di un gesto o movimento da parte dell'interlocutore, ad esempio un cenno della testa;
- *Backchannel* - con questo termine si indica un evento, un segnale linguistico o gestuale, che l'interlocutore fa, durante una conversazione, per segnalare all'altra persona di "essere sul pezzo". Esempio di questo evento è il tipico "uh, huh" di approvazione.[8]

Nonostante, come si è visto, in letteratura si possano trovare differenti definizioni e interpretazioni del fenomeno considerato, molti ricercatori concordano, comunque, nel considerarlo un fenomeno dalle molteplici sfaccettature[9]: un fenomeno, cioè, che comprende molti aspetti, alcuni più facilmente osservabili, altri meno, che, però, contribuiscono tutti, in egual misura, alla sua definizione.

3.1.1 Coinvolgimento e Attenzione

Una volta compreso cosa si intende per *engagement*, l'altro aspetto da affrontare è la questione della resa in italiano del termine.

Traduzioni letterali lo renderebbero come *impegno* o *coinvolgimento* (dunque come sinonimi dei corrispettivi inglesi *commitment* e *involvement*). *Coinvolgimento* è, certamente, una parola che riesce a dare conto degli aspetti generali del fenomeno in questione; tuttavia, nella letteratura, un altro termine inglese, si è fatto strada: *attention*, la cui traduzione in italiano sarebbe *attenzione*.

Goldberg et al. (2019) definiscono l'attenzione come una componente del *behavioral engagement*, che, si ricorda, è una tipologia di *engagement* visibile

[7] «Engagement is the process by which interactors start, mantain and end their perceived connection to each other during an interaction».

[8] Rich et al. (2010, 376-sgg.).

[9] Cfr. Fredricks, Blumenfeld e Paris (2004, p. 60): «the multifaceted nature of engagement is also reflected in the research literature» («la natura variegata dell'engagement si riflette anche nella letteratura di riferimento»).

all'esterno in quanto più strettamente ha a che fare con il comportamento di colui che ascolta. Essendo l'*engagement*, di per sé, un fenomeno estremamente difficile da studiare per le sue numerose componenti non osservabili, l'*attention* è divenuto un oggetto di studio privilegiato da parte di numerose ricerche. Tale ragione ha contribuito ad estendere la definizione di *attention* fino a ritenerla una componente visibile dell'*engagement* a tutto tondo. Non è un caso, perciò, che in letteratura questi due termini vengano spesso sovrapposti e usati come sinonimi per indicare lo stesso fenomeno.

In questa sede si tenderà a usare prevalentemente la terminologia più specifica di *attenzione* insieme al suo correlato inglese *attention*. Tuttavia, sarà possibile ritrovare anche l'uso del termine *engagement* che, in tal caso, dovrà essere inteso come sinonimo di attenzione, in quanto fenomeno oggetto del nostro interesse.

3.2 Lo studio dell'attenzione in un corpus multimodale di visite turistiche

Avendo la possibilità di lavorare su un corpus multimodale contenente al suo interno dati relativi a visite turistiche guidate, l'Istituto di Linguistica Computazionale di Pisa ha ben pensato di condurre su di esso un interessante studio relativo al fenomeno di cui sopra si è largamente discusso. In particolare, il quesito che ha mosso l'interesse dei ricercatori è stato il seguente: il livello di coinvolgimento mostrato dai partecipanti durante la visita guidata potrebbe essere correlato ad alcune feature linguistiche e/o prosodiche del discorso della guida turistica?

Per poter rispondere a questa domanda si rendeva, innanzitutto, necessario annotare il livello di attenzione mostrato dai visitatori nei confronti dei contenuti presentati dalla guida. Annotare una tale informazione era possibile in virtù del fatto che si possedeva, come già evidenziato nel paragrafo 2.3, di un'inquadratura sull'audience di visitatori.

Per tale ragione sono stati promossi e avviati, nell'a.a. 2019/2020, due progetti di tirocinio, svolti rispettivamente da due colleghi Mario Gomis e Luca Poggianti, entrambi laureandi del Corso di Laurea triennale in Informatica Umanistica dell'Università di Pisa. Scopo dei tirocini era quello di procedere all'annotazione visiva dell'*attention* mostrata dal pubblico durante le visite guidate presso la Certosa di San Martino.

Entrambi i tirocini si sono, inoltre, tradotti in due lavori di Tesi, uno dei quali volto all'identificazione, in un corpus ristretto di dati, delle possibili feature linguistiche determinanti il livello di coinvolgimento del pubblico.

Nelle prossime sezioni si farà riferimento solo ed esclusivamente a quest'ultimo lavoro a cui si è dedicato Poggianti e al quale, ovviamente, si rimanda per ulteriori chiarimenti e/o approfondimenti.[10]

[10] Poggianti (2020).

3.2.1 Il sub-corpus di dati CHROME

Prima di addentrarsi a parlare di come i dati siano stati annotati durante le esperienze di tirocinio di cui sopra si parlava, è necessario precisare che l'ILC si è occupato solo di un sottoinsieme di dati provenienti dal progetto CHROME. Se le registrazioni audio-video delle visite guidate presso la Certosa di San Martino riguardavano 3 diverse guide turistiche, l'Istituto di Linguistica Computazionale ha acquisito i dati relativi a una sola guida che accompagna 4 gruppi di visitatori attraverso tutti gli ambienti del complesso monastico napoletano.

Ognuna di queste informazioni è desumibile dal nome dei file utilizzati in fase di annotazione, che possono avere, ovviamente, formati differenti a seconda che si tratti di registrazioni video, audio o materiali testuali:

- G01: indica il codice della guida turistica che, come detto, rimane sempre la stessa;
- V01-2-3-4: indica il codice della visita;
- P01-2-3-4-5-6: indica il codice associato a uno specifico luogo di interesse.

Per meglio esemplificare: la stringa alfanumerica G01V03P05 indicherebbe un dato relativo alla visita guidata n° 3, in particolare all'ambiente n° 5 (il Coro), e presieduta dalla guida turistica 1.

Oltre alle registrazioni audiovisive, sono state, infine, acquisite dall'ILC anche le trascrizioni del parlato della guida relative alle prime 3 visite e al solo POI 1, ovvero il Pronao.

Figura 3.1: I dati CHROME acquisiti dall'ILC.

Per avere un'idea immediata della tipologia di dati CHROME acquisiti dall'ILC si faccia riferimento alla Figura 3.1.

3.2.2 Annotazione dell'*attention*

Annotare l'attenzione che i visitatori ponevano nei confronti delle informazioni veicolate dalla guida turistica nel corso della visita non era affatto semplice per

via della molteplicità di indicatori, designanti la presenza o assenza del fenomeno in questione, che avrebbero reso la sua valutazione estremamente soggettiva.

L'impossibilità di interpretare univocamente il fenomeno di *attention* ha avuto tre importanti conseguenze:

1. La scelta di affidare il compito di annotazione a due annotatori umani. L'impossibilità di individuare dei criteri oggettivi per l'individuazione del fenomeno in questione rendeva, infatti, difficile programmare e sviluppare dei sistemi di annotazione automatica;

2. La scelta di fornire agli annotatori umani, se non delle regole categoriche, quanto meno delle linee guida da tenere in considerazione durante il trattamento dei dati;

3. La scelta di effettuare un'analisi di correlazione tra i due set di annotazione prodotti per determinarne l'affidabilità.

Come si evince da Poggianti (2020), i parametri discriminanti di cui si è maggiormente tenuto conto sono stati: lo sguardo dei visitatori, i loro gesti e movimenti, e il loro grado di partecipazione attiva alla visita (che prevede, ad esempio, la manifestazione di dubbi, domande o curiosità).

L'annotazione è stata effettuata mediante una piattaforma dedicata: PAGAN (*Platform for Audiovisual General-purpose ANnotation*),[11] che permette all'utilizzatore di segnalare l'aumento o l'abbassamento di un certo fenomeno di interesse semplicemente premendo le freccette su-giù della tastiera. Una volta completata l'annotazione, lo strumento fornisce in output un file csv il cui contenuto sarà analizzato più in dettaglio nel paragrafo 4.2.7.

Nonostante gli annotatori abbiano lavorato in maniera del tutto indipendente, confrontando i risultati e calcolando l'*inter-annotator agreement*, si sono ottenuti dei risultati di correlazione piuttosto alti (la maggior parte dei valori si attesta sull'ordine dello 0.80 e 0.90)[12], segno che l'annotazione è avvenuta secondo criteri, per così dire, "universali" e non sulla base di interpretazioni soggettive che avrebbero reso meno solide le considerazioni finali dei progetti di Tesi dei due tirocinanti.

A conclusione di questo lavoro sono stati, dunque, prodotti dei set di dati relativi al fenomeno di *attention*, dei quali si è, effettivamente, dimostrata la non randomicità confermando la possibilità di renderli una buona base per eventuali sviluppi futuri.

3.2.3 La prosecuzione dei lavori

L'annotazione dell'*attention* nel sub-corpus di dati CHROME ha costituito solo la prima parte di un tirocinio svolto da Poggianti che ha previsto anche, come

[11] Lo strumento è consultabile al seguente indirizzo web: `https://pagan.institutedigitalgames.com/index.php`. Per informazioni più approfondite si rimanda, invece, a Melhart, Liapis e Yannakakis (2019).

[12] Il coefficiente di correlazione prescelto per la misurazione è stato il *coefficiente di correlazione per ranghi di Spearman*. Per una trattazione esauriente di come sia avvenuto il calcolo di *agreement* si consulti Poggianti (2020).

accennato prima, una fase di profilazione linguistica del materiale testuale al fine di identificare eventuali feature collegate al fenomeno di attenzione.

Per far questo, un ruolo centrale hanno assunto ovviamente le trascrizioni del parlato della guida turistica. Dal momento che si trattava di testi rappresentanti, di fatto, delle produzioni orali, essi presentavano al loro interno, oltre che la normale successione delle parole, dei tag utili a segnalare tutti i fenomeni propri della lingua parlata come pause, interruzioni, ripensamenti, allungamenti vocalici, risate, inspirazioni, schiocchi di lingua, etc.

Una lista completa dei tag utilizzati, con i rispettivi fenomeni associati, può essere ritrovata in un documento (Savy 2006) prodotto in seno al progetto CLIPS, che, negli anni 1999-2004, ha avuto come scopo primario l'individuazione di strumenti per lo studio e il trattamento automatico dell'italiano, sia scritto ma soprattutto parlato.

A partire dalle trascrizioni, il testo dei POI di ogni visita è stato scomposto e suddiviso in frasi adoperando come delimitatore di frase le pause, brevi e lunghe, rispettivamente indicate dai tag <sp> (*short pause*) e <lp> (*long pause*). Ovviamente, una simile scomposizione non assicurava che i segmenti risultanti costituissero delle frasi nel senso proprio in cui le intenderebbe un parlante nativo di una lingua.[13] Eppure, per gli scopi, per così dire, "esplorativi" dell'analisi condotta, questa strategia si è rivelata sufficiente.

Dopo aver ripulito il testo da tutti i tag superflui, per ogni frase, sono state estratte le feature linguistiche considerate rilevanti per la lingua parlata. La decisione è stata quella di mettere da parte le caratteristiche strettamente sintattiche per focalizzarsi solo su:

- Numero di tokens per frase;
- Numero medio di caratteri per token;
- Presenza percentuale dei vari POS tag (*Part Of Speech Tag*).

Lo strumento usato per questa fase di profilazione è *Profiling-UD*, lo stesso di cui si parlerà anche nella sezione 4.2.3.1, pertanto se ne rimanda la trattazione. Le feature risultanti sono state, infine, opportunamente esaminate con l'ausilio del test statistico *U di Mann-Whitney* al fine di decretare la reale influenza di ciascuna feature nella definizione di una delle due popolazioni in questione: insieme delle frasi positivamente annotate e insieme delle frasi negative o neutre.

In conclusione, ciò che è emerso è che feature come numero di token per frase e presenza di avverbi, ausiliari, congiunzioni coordinanti, pronomi, nomi propri, congiunzioni subordinanti e verbi, influenzano il grado di attenzione che i visitatori pongono nei confronti del discorso di una guida turistica.

Gli ottimi risultati rivelati da questi primi studi sull'*attention* hanno alimentato il desiderio di condurre un'analisi più accurata e approfondita che, da un lato, aggiungesse più assi di descrizione dei dati (ad esempio l'aggiunta, per ogni segmento, delle feature acustiche) e che, dall'altro, superasse alcuni dei

[13] Il concetto di frase è estremamente difficile da definire. Di ciò si parlerà più approfonditamente *infra* nel paragrafo 4.2.1.

limiti caratterizzanti i lavori di Gomis e Poggianti: *in primis* il problema riguardante la segmentazione delle trascrizioni del parlato in unità di base che potessero davvero dar conto di un "parlato scritto". Difatti le pause, brevi o lunghe che siano, non possono essere considerate dei separatori affidabili per l'individuazione delle minime unità scomponibili del parlato. Una delle ragioni è, per esempio, quella per cui spesso, in un discorso, le pause hanno la funzione di "far prendere fiato" a colui che parla: sarebbe, infatti, improbabile riuscire a pronunciare un'intera frase del discorso senza i dovuti respiri. Così come è vero il contrario: il parlante potrebbe anche decidere di non mettere alcuna pausa nel discorso, laddove, invece, ci vorrebbe (specialmente se il discorso fosse scritto).

Capitolo 4

L'esperienza di tirocinio

L'obiettivo di questo capitolo è descrivere la prima parte dell'esperienza di tirocinio, svoltasi tra i mesi di marzo e luglio 2021 presso l'Istituto di Linguistica Computazionale di Pisa, e che ha avuto come scopo primario la costruzione di un corpus italiano di frasi di parlato relativo al dominio delle visite guidate.

L'idea è stata, poi, quella di arricchire questo corpus di informazioni linguistiche, acustiche, tematiche e relative al fenomeno di *attention* (di cui molto si è parlato nel precedente capitolo).

Si affronteranno, perciò, i diversi problemi con i quali ci si è scontrati cercando di spiegare quali ragionamenti siano stati fatti per affrontarli e risolverli.

4.1 Nuovi progetti di tirocinio presso l'ILC

Per riassumere quanto esposto nei capitoli 2 e 3, l'Istituto di Linguistica computazionale "Antonio Zampolli" di Pisa, a conclusione del progetto CHROME, si è fatto promotore di una serie di studi relativi al fenomeno di *attention* che si può, in maniera molto rapida, definire come l'interesse che gli uditori pongono nei confronti del loro interlocutore.[1]

Gli ottimi risultati mostrati a seguito dei lavori di Gomis e Poggianti, di cui si è brevemente discusso nel capitolo 3, hanno stimolato i ricercatori dell'Istituto a condurre delle indagini più approfondite in merito al tema in questione.

Innanzitutto, la prima questione da risolvere era relativa alla necessità di individuare unità alla base del parlato appropriate per un'analisi condotta mediante l'uso di strumenti di profilazione linguistica tarati per lo scritto. Anche in questo caso, dunque, l'ILC ha promosso due progetti di tirocinio svolti rispettivamente da me e da un collega della laurea triennale in Informatica umanistica, Federico Boggia.

[1] Per una trattazione più esauriente e completa si rimanda *supra*, par. 3.1.

A entrambi sono stati forniti i seguenti formati di dati (la cui natura e struttura è stata ampiamente definita nei capitoli 2 e 3)[2]:

1. Tracce audio in mp4;
2. Trascrizioni allineate del discorso della guida in csv;
3. File eaf (di cui si comprenderà la natura nella sottosezione 4.2.1.1).

Il compito a noi assegnato ha previsto l'ascolto delle varie tracce audio fornite e la conseguente modifica dei file csv riportando, in essi, la corretta segmentazione in frasi del parlato della guida turistica. Di questa delicatissima fase di processing del dataset si parlerà molto più dettagliatamente nel paragrafo 4.2.1.

Se la prima differenza con lo studio svolto dai due precedenti tirocinanti risiede nella qualità dell'annotazione prodotta, la seconda riguarda, invece, da un lato, la quantità dei dati presi in esame e, dall'altro, la quantità delle feature descrittive dei dati stessi.

Lo studio di Poggianti analizzava, infatti, 3 esposizioni diverse della guida fatte, tuttavia, esclusivamente nel Pronao (cfr. Figura 2.3). I file da noi analizzati hanno, invece, riguardato tutti i POI delle 3 visite in questione. In particolare, sono state da me annotate le visite 2 e 3, e da Boggia le visite 1 e 3. Avere tra le mani un'annotazione comune (quella della visita 3) avrebbe dato la possibilità di effettuare un controllo della validità della segmentazione prodotta attraverso il calcolo dell'*agreement* tra i due annotatori, come verrà spiegato più in dettaglio nel paragrafo 4.2.2.

Inoltre, i dati di Poggianti sono stati annotati esclusivamente con delle informazioni linguistiche. Nel presente lavoro (così come in quello di Boggia), oltre alla profilazione linguistica, si è proceduto anche all'annotazione di informazioni prosodico-acustiche, di cui si rimanda la trattazione nel paragrafo 4.2.5.

Una delle differenze che, invece, caratterizzano il mio lavoro rispetto al recentissimo studio di Boggia è l'aggiunta di informazioni, per così dire, tematiche,[3] la cui presenza dà la possibilità di usare il dataset finale per uno studio differente rispetto a tutti quelli finora condotti presso l'ILC e volto all'identificazione di una possibile relazione tra il piano linguistico e fonetico-acustico da un lato e il piano semantico dall'altro, come verrà dettagliatamente spiegato nel capitolo 7.

Ovviamente, come in tutte le ricerche, era impossibile prevedere l'esito delle nostre analisi, seppure quelle precedenti avessero mostrato risultati interessanti e favorevoli alla prosecuzione delle indagini. In altre parole, era impossibile avere, a priori, una certezza assoluta circa l'effettiva possibilità di individuare dei pattern ricorrenti che avrebbero potuto porsi alla base del fenomeno di attenzione o, più genericamente, di un qualsivoglia fenomeno linguistico-comunicativo.

In questa trattazione non si farà riferimento ai risultati ottenuti nello studio del collega Federico Boggia, al quale si rimanda caldamente per ulteriori approfondimenti in materia.[4]

[2] In particolare si rimanda *supra* alla sezione 2.5 per una descrizione più alto livello dei dati CHROME e al paragrafo 3.2.1 per il sotto-insieme di dati acquisiti, in specifico, dall'ILC.

[3] Cfr. *infra*, par. 4.2.6.

[4] Cfr. Boggia (2021).

4.2 La costruzione del dataset

Le operazioni di manipolazione che sono state fatte sul dataset sono molteplici e, ognuna, con una propria rilevanza. Le sezioni che verranno hanno lo scopo di fornire un'idea più chiara e dettagliata possibile della pipeline seguita per la messa a punto di un dataset adatto per l'addestramento di due classificatori: da un lato un classificatore di *attention* e dall'altro un modello per la previsione di informazioni tematiche,[5] come si avrà modo di approfondire, rispettivamente, nei capitoli 6 e 7.

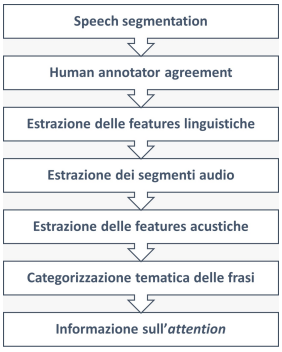


Figura 4.1: Schema riassuntivo degli step seguiti per la costruzione del dataset finale.

4.2.1 *Speech segmentation*

La parte cosiddetta di *sentence splitting* (o *speech segmentation*) è stata, probabilmente, la parte più delicata di manipolazione del dataset, non solo per la complessità che la caratterizza ma soprattutto perché, su questo primo step, si sono fondati la totalità di quelli successivi.

La necessita di affrontare il problema relativo alla suddivisione in frasi del parlato della guida turistica era, in parte, legata agli strumenti che si sono utilizzati nella fase di annotazione del dataset. La profilazione linguistica, ad esempio, (di cui si parlerà in dettaglio nel paragrafo 4.2.3) è stata fatta tramite *Profiling-UD*, uno strumento di annotazione che nasce per la lingua scritta e non parlata. Avere, dunque, un corpus di lingua parlata rappresentato con le convenzioni tipiche dello scritto era un prerequisito fondamentale per una corretta estrazione delle informazioni linguistiche dal testo.

Tuttavia, sarebbe impossibile condurre delle analisi linguistiche senza prima mettere per iscritto le registrazioni di parlato a disposizione. Come si legge in Izre'el et al. (2020), difatti, il più grande paradosso della linguistica è proprio la difficoltà con la quale si riesce a studiare quella che dovrebbe essere il suo oggetto di studio primario e privilegiato, ovvero la lingua parlata, che però deve

[5] Per capire di che tipologia di informazione si tratti si consulti *infra*, par. 4.2.6.

necessariamente essere riportata in una varietà diamesica e una forma differente rispetto a quella in cui intrinsecamente nasce.

Nel nostro caso, il problema risiedeva nel suddividere, a partire dai file audio, il flusso del parlato al fine di rappresentarlo sotto forma di frasi, un task tutt'altro che banale.[6] I parlanti nativi di una lingua, infatti, quando elaborano o ascoltano un discorso, non pensano a dove inizierà o terminerà una frase: semplicemente si immergono all'interno del flusso di parole, con l'unico scopo di veicolare o captare un certo significato.

In altre parole, il concetto di frase, così come lo si intende nella lingua scritta di "segmento individuabile e solitamente compreso tra due punti fermi",[7] non esiste nella lingua parlata a meno che non lo si introduca più o meno forzatamente. Se si provasse, cioè, a chiedere a un parlante di segmentare, a partire da un audio, il corrispondente testo, dopo non molto, fatte una serie di riflessioni e applicati alcuni criteri,[8] egli riuscirà a individuare dei segmenti che tanto assomigliano al familiare concetto di frase che tutti noi, da parlanti adulti della nostra lingua, conosciamo.[9] Ciò è possibile in virtù del fatto che, in realtà, esiste una definizione prosodica di "frase del parlato": tutto quello che il parlante dovrà fare è, difatti, segmentare il testo sulla base del *pitch* intonativo e mettere conseguentemente i punti all'interno della trascrizione di quello specifico parlato.[10] Se nella lingua scritta i punti sono i demarcatori che visivamente indicando dove una frase termina e l'altra comincia, nella lingua parlata sembrerebbe, invece, la prosodia a determinare l'inizio e la fine di una frase, come si legge in Izre'el et al. (2020).

Ovviamente la prosodia sarebbe solo uno degli aspetti da considerare per individuare i segmenti alla base della lingua parlata. Un altro aspetto che sembrerebbe contribuire è quello semantico. Un parlante ha, infatti, la percezione di frase quando un enunciato è dotato di senso compiuto e può essere isolato dai contesti circostanti.

Nonostante sia estremamente difficile individuare le regole alla base della segmentazione del parlato, l'individuazione delle unità alla base della lingua parlata, da parte di un parlante nativo, dovrebbe, comunque, risultare più o meno naturale in quanto si tratta di un fattore prevalentemente percettivo. Da qui, appunto, il senso dei lavori di segmentazione svolti da me e dal collega Federico Boggia.

[6] Come più volte si è detto, il lavoro di Poggianti si era basato su una segmentazione automatica del discorso della guida turistica basata sull'individuazione dei tag di pausa, i quali però non cadono necessariamente in corrispondenza dei confini sintattici. Un'annotazione fatta in tal senso non era, dunque, in grado di tenere conto dei molteplici fenomeni caratterizzati la lingua parlata, come pause, brusche interruzioni di frasi, riformulazione del discorso, etc.

[7] Questa è la definizione intuitiva che darebbe un qualsiasi parlante della lingua italiana. Ovviamente la percezione di frase muta a seconda della natività del parlante stesso.

[8] Per avere un'idea di quali siano questi criteri si consulti *infra*, sottosezione 4.2.1.2.

[9] È interessante osservare come molti dei concetti alla base dello studio della lingua siano concetti che si conoscono inconsciamente e di cui si può difficilmente fornire una definizione precisa.

[10] Solitamente un *pitch* ascendente è associato alla volontà di un parlante di continuare una frase; al contrario, un *pitch* discendente è associato alla volontà di terminarla.

A grandi linee, il criterio che si è seguito per la fase di *sentence splitting* è molto semplice e può essere suddiviso in due step:

1. Ascolto del file audio e individuazione dei possibili gruppi di parole analizzabili come singola unità;
2. Sulla base delle caratteristiche prosodico-intonative e semantiche, si stabilisce l'effettivo status del gruppo di parole individuato. Se l'analisi ha riscontro positivo, si passa all'individuazione del segmento successivo, altrimenti si modifica il gruppo di parole selezionato al fine di individuare una nuova unità eleggibile allo status di frase.

Nel primo step, grande importanza ha assunto l'utilizzo di ELAN, un programma per l'annotazione multimodale di cui si parlerà nel prossimo sottoparagrafo. Del secondo step, e dunque dei criteri utilizzati per la definizione di ogni singolo segmento, si parlerà, invece, più in dettaglio, nella sottosezione 4.2.1.2.

4.2.1.1 ELAN: un software per l'annotazione multimodale

Si è più volte parlato nei capitoli precedenti della multimodalità che caratterizza non solo il progetto CHROME ma anche tutti i progetti che da quest'ultimo derivano.

Per effettuare delle indagini che coinvolgessero più livelli di analisi, si è visto necessario usare uno strumento per l'annotazione multimodale in grado di visualizzare i diversi strati di informazione (nel nostro caso l'informazione linguistica del testo allineata con l'audio) in maniera del tutto *user-friendly*: ELAN.[11] È proprio questo programma che, una volta processato un file, ne fornisce in output uno con estensione eaf (ovvero *ELAN Annotation Format*).

Come accennato, nel caso del nostro studio, si è reso necessario analizzare parallelamente il livello acustico e linguistico. L'allineamento di queste informazioni ha reso possibile una maggiore accuratezza nella fase di segmentazione del testo in quanto, a partire dall'elenco di parole contenuto nella sezione *Grid* (Cfr. Figura 4.2), è stato possibile selezionare insiemi distinti di token e ascoltarne il corrispondente segmento audio. L'ascolto del singolo segmento, isolato da tutto il restante contesto linguistico-acustico, ha avuto come scopo quello di valutare al meglio le caratteristiche dell'insieme al fine di smentire o confermare la possibile eleggibilità del segmento allo status di frase.

Inoltre una vista dell'onda sonora ha reso molto più fine e scrupoloso il processo di individuazione, in millisecondi, degli start ed end di frase, controllo che pure è stato effettuato in quanto necessario per l'individuazione dei segmenti audio dai quali sono state, poi, estratte le feature acustiche.[12]

Per riassumere, in questo studio, ELAN ha costituito un supporto per la fase di *speech segmentation*, consentendo una facile individuazione non solo dei gruppi di parole che era possibile considerare delle frasi, ma anche degli start e degli end di ogni singolo segmento.

[11] Sito di riferimento: `https://archive.mpi.nl/tla/elan`.
[12] Cfr. *infra*, par. 4.2.5.

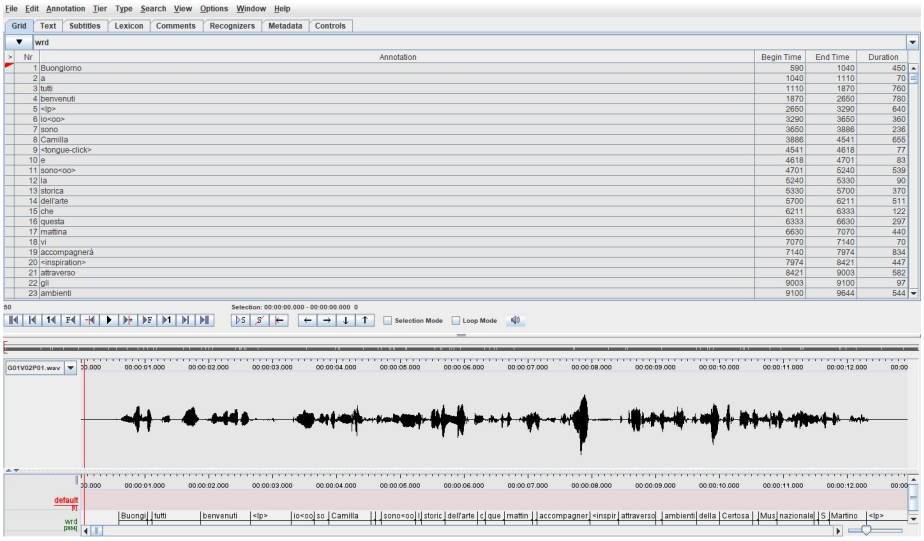


Figura 4.2: Vista del software ELAN (G01V02P01).

4.2.1.2 I criteri applicati e le convenzioni grafiche per l'annotazione

Per cercare di mantenere il più naturale possibile il processo di segmentazione in frasi del parlato, la scelta è stata quella di non fornire agli annotatori umani delle regole categoriche da rispettare, ma lasciare che questi ultimi si lasciassero guidare dalle informazioni prosodico-semantiche, di cui prima si parlava, nonché dall'intuizione e dalla conoscenza intrinseca di cui ogni parlante è dotato in merito alla propria lingua.

Prima dell'assegnazione del task sono state fornite a me e al collega Federico Boggia delle linee guida sugli aspetti sui quali porre una maggiore attenzione per determinare il confine tra le frasi. Di ognuno dei criteri impiegati si fornirà una panoramica qui di seguito, cercando anche di evidenziare la logica applicata al fine di garantire la consistenza dell'annotazione stessa.

Si è, innanzitutto, puntato all'individuazione di un pitch intonativo discendente. Come anche detto nel paragrafo 4.2.1, una frase si ritiene conclusa quando la si chiude intonativamente. Questa regola, come le altre che verranno esposte, presenta numerose eccezioni. Ad esempio, la frase 4.2.1 è ascendente (nella Figura 4.3 si nota che il pitch di chiusura è relativamente alto) ma, al contempo, si ha la percezione che essa sia, semanticamente, terminata. In tal caso, i due segmenti sono stati separati.

Esempio 4.2.1. Il suo vero nome era Domenico Gargiulo, ma per gli amici, noi lo chiamiamo Micco (G01V02P04)

Il secondo criterio di cui si è tenuto conto è stata la variazione del topic e l'eventuale presenza di una frase semanticamente indipendente. In altre parole,

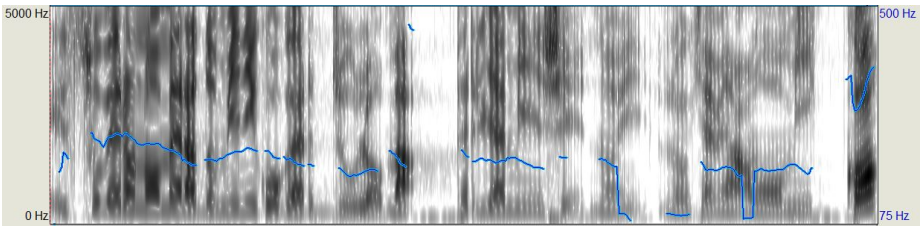


Figura 4.3: Spettrogramma (in grigio) e pitch intonativo (linea blu) della frase contenuta nell'esempio 4.2.1.

due enunciati vengono separati se il focus primario del discorso viene modificato, dando, dunque, un senso di compiutezza e indipendenza al segmento.

Altro importante aspetto preso in considerazione nella segmentazione in frasi del discorso è stata la presenza di pause molto evidenti. Una pausa marcata suggerisce, infatti, la fine di una frase, anche nell'eventualità in cui quest'ultima sembri continuare dal punto di vista dell'intonazione. Una pausa durante un discorso, dà, infatti, al parlante, l'impressione che vi sia quello che, nella lingua scritta, corrisponderebbe a un punto, dunque a un marcatore di fine frase.

Esempio 4.2.2. Siamo quindi passato da un ambiente all'altro. Vi invito come sempre a guardarvi intorno indipendentemente da quello che io poi vi descrivo nel dettaglio (G01V02P04)

Tra il pronome "altro" e il pronome riflessivo "vi" intercorre un tempo di 1863 millisecondi, che è un intervallo relativamente alto (dato che si tratta di una pausa di quasi 2 secondi nel bel mezzo di un discorso). Bisogna, tuttavia, applicare il criterio appena esposto considerando i singoli e particolari casi, in quanto non è sempre vero il contrario: l'assenza di una pausa non garantisce, infatti, che la frase continui. Si è notato, ad esempio, che spesso la guida sembra legare tra loro due frasi per la semplice ragione di aver effettuato, nel corso della frase precedente, numerose pause che le consentono di proseguire, spedita e senza sosta, nella formulazione della frase successiva. In tal caso si è deciso di individuare questa tipologia di enunciati e di suddividerli, seppure a livello prosodico-intonativo il discorso sembri fluire ininterrotto. Un esempio di questo caso si riporta qui di seguito.

Esempio 4.2.3. Quindi sono tutte quante le virtù che il buon certosino dovrebbe avere. In particolare coloro che poi sono, di fatto, a capo della Certosa (G01V02P04)

Nonostante si noti, in Figura 4.4, l'assenza totale di una pausa tra le due frasi riportate nell'esempio precedente (il confine tra le due è segnalato dal punto nell'esempio e dalla riga rossa tratteggiata nell'immagine), esse sono state separate. Si presume, infatti, che la guida, avendo effettuato molte pause in chiusura della prima frase (si vedano i riquadri in giallo nella Figura) sia in grado di iniziare la seconda senza "prender fiato".

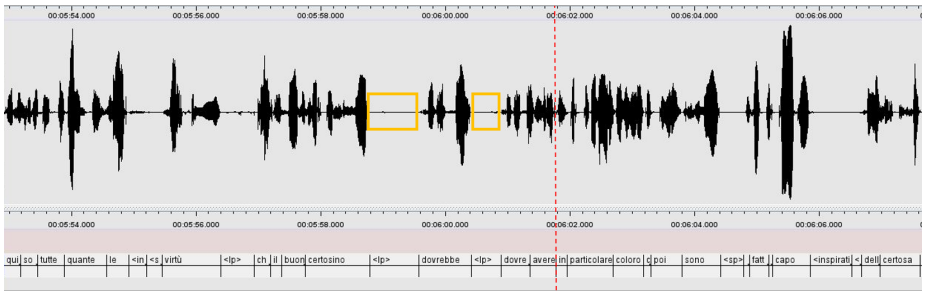


Figura 4.4: Onda sonora della frase riportata nell'esempio 4.2.3.

Infine, si è guardato alla forza intonativa di chiusura di una frase. Questo criterio potrebbe essere confuso con il precedente; tuttavia, esso guarda non tanto alla presenza di un'eventuale pausa di fine frase, quanto alla marcatezza prosodica dell'ultima parola. Se la guida conclude una frase con molta forza (rendendo quasi l'idea di mettere un punto), il gruppo di parole successivo viene separato dal precedente. Viceversa, in maniera del tutto complementare, si è tenuto in considerazione il caso opposto, ovvero di un'eventuale debolezza dell'inizio della frase successiva: se l'attacco alla nuova frase era privo di forza intonativa, si è ritenuto opportuno mantenere uniti i due segmenti.

Il caso ottimale si verificava quando due o più condizioni occorrevano simultaneamente (es. pausa evidente e cambiamento di topic). È da tenere presente però che i casi, per così dire, ideali costituivano l'eccezione piuttosto che la regola, rendendo, appunto, la fase di *speech segmentation* non sempre facile da gestire né priva di ambiguità, anche per il carattere di imprevedibilità che caratterizza una varietà della lingua quale quella parlata.

Inoltre, da sottolineare che i criteri sopra esposti sono quelli che maggiormente si sono applicati, ma che di certo non possono tener conto di tutti i possibili casi che si sono, di volta in volta, incontrati. Un grande problema hanno costituito, per esempio, molte frasi relative che spesso sembravano legate alla proposizione principale, ma che poi, per motivi legati al fattore "lunghezza" di una singola frase, è stato necessario dividere.[13]

[13]Per evitare un forte sbilanciamento in termini di lunghezza delle frasi, è stato spesso necessario segmentare le frasi anche laddove la guida turistica non dava segni di pause, interruzioni, cambiamenti di topic e così via. Un forte squilibrio avrebbe, infatti, alterato il processo di estrazione delle feature acustiche: frasi composte, ad esempio, semplicemente da un "ok" e frasi composte, invece, da un gran numero di token avrebbero comportato delle grosse differenze in termini di valori delle feature acustiche estratte. Normalmente, nella lingua scritta, le frasi relative appartengono alla stessa "frase" della proposizione reggente. Nel parlato le cose stanno diversamente, dal momento che le enunciazioni potrebbero essere anche molto lunghe in virtù di quella che viene definita *programmazione online del parlato*: un parlante può, cioè, più volte riformulare il discorso o, addirittura, generare frasi anche molto lunghe, aggiungendo all'infinito tutte le informazioni che decide di veicolare. La tendenza è stata quella di evitare di avere troppi squilibri nella lunghezza (misurata in termini di token per frase). Ciò si è deciso per rendere più chiaro il processo di estrazione delle feature acustiche.

Per poter procedere con la segmentazione in frasi del testo è stato, inoltre, necessario stabilire una serie di convenzioni grafiche. In particolare, ci si è avvalsi:

- Del punto come demarcatore tra le frasi;
- Della virgola per segnalare la presenza, entro una stessa frase, di proposizioni parentetiche, elenchi o pause logiche;
- Delle parentesi tonde seguite da + per segnalare porzioni di testo da epurare in un secondo momento in quanto contenenti disfluenze.[14]

4.2.1.3 Alcune riflessioni

Durante questa prima fase di processing del dataset, in cui si è reso necessario ascoltare, accuratamente e più di una volta, registrazioni audio di lingua parlata, mi sono ritrovata a riflettere su quali siano o possano essere le abitudini linguistiche di parlanti che si ritrovino a interagire con un pubblico più o meno ampio di ascoltatori,[15] e dunque in tutti quei casi in cui ci si trova in un contesto di *parlato programmato* o *semi-programmato* (nel senso che il parlante sa, a linee generali, quali sono i contenuti da trasmettere, senza però aver imparato il suo discorso a memoria).

Seppure le mie riflessioni siano scaturite dall'ascolto di un particolare tipo di parlato e potrebbero, quindi, essere state influenzate dalle caratteristiche diastratiche della guida turistica (es. il fatto di essere una donna, con un certo livello di cultura, e probabilmente campana/napoletana, dato il suo forte accento), è lecito pensare che alcuni fenomeni siano, in un certo qual modo, universali. Per averne conferma, basta semplicemente porre una maggiore attenzione a tutti quei casi di comunicazione uno a molti a cui assistiamo quotidianamente o, banalmente, andando oltre lo specifico *case study*, alla stragrande maggioranza delle ordinarie interazioni tra due o più persone.

In generale, si registrerà la ricorrenza dei seguenti fenomeni:

1. Tendenza a interrompere quello che si sta dicendo, per riformularlo o, addirittura, cambiare totalmente il focus del discorso. Un esempio si ritrova nel file G01V03P06:

   Esempio 4.2.4. (Se qualcuno di voi l'avesse visto, una delle parti del film è proprio girata)+ Forse addirittura si apre proprio con una scena girata qui...

   La frase segnalata con il simbolo ()+[16] in 4.2.4 rimane aperta, quasi incompleta, in quanto la guida modifica, subito dopo, la struttura attraverso cui veicolare il messaggio.

[14] Per la definizione di disfluenza si rimanda *infra*, sottosezione 4.2.1.3. Della fase di pulizia del testo si parla, invece, *infra*, par. 4.2.2.

[15] Come sottolineato *supra* (capitolo 1, nota 2), in questo lavoro si sta considerano un tipo di comunicazione specifico, ovvero uno a molti.

[16] Il simbolo ()+ viene utilizzato per segnalare le disfluenze nel testo. Cfr. *supra*, sottosezione 4.2.1.2.

2. Tendenza a "prender tempo" utilizzando quelle che, in gergo tecnico, vengono definite *pause piene*. Queste pause vengono solitamente fatte allo scopo di elaborare il contenuto di ciò che verrà detto o di scegliere la struttura sintattica più adeguata. Vengono, inoltre, così chiamate in quanto i vuoti delle pause vengono riempiti da piccoli suoni (ad esempio: "ehm", "mmm") o dalla ripetizione di parole appena pronunciate come in questo caso:

   Esempio 4.2.5. Come sempre ci sono gli stalli, (ve lo avevo)+ ve lo avevo anticipato (G01V02P04)

3. Tendenza ad avere dei "tic linguistici", ovvero parole che si ripetono spesso durante il discorso. Nel caso della guida turistica selezionata per questo studio, ad esempio, ricorrono spesso il verbo "diciamo" e avverbi come "insomma" e "appunto". Queste parole vengono spesso usate come riempitivi, ricadendo, dunque, nel caso precedente.

Il primo e il secondo punto rientrano in quella che, tecnicamente, viene definita *disfluenza*. Con questo termine si indica, genericamente, una qualsiasi interruzione del naturale scorrere del parlato che si può, appunto, manifestare sotto forma di interruzioni, riformulazioni oppure di ripetizioni.

Interessante è, inoltre, notare come un discorso possa fluire più o meno speditamente a seconda che si tratti di quello che sopra si è definito *parlato programmato* o no. La guida di cui si è analizzato lo *speech* è, ad esempio, in certi punti del suo discorso, molto più sicura e i momenti di esitazione meno frequenti.[17] In altri punti, invece, come ad esempio a seguito di una domanda da parte di uno dei visitatori, la guida si ritrova a "improvvisare" elaborando quasi a fatica un discorso.

Ovviamente riflessioni di questo tipo vanno oltre gli scopi della presente trattazione. Tuttavia, dal momento che molto si sta parlando di comunicazione e, in particolare, di una comunicazione di cui i principali protagonisti sarebbero agenti artificiali (si vedano gli scopi di CHROME nel capitolo 2) sarebbe interessante rispondere alla seguente domanda: per rendere queste macchine più umane nel loro modo di comunicare, sarebbe il caso di dotarle di questi meccanismi di disfluenza, che sono fenomeni linguistici tipicamente e propriamente umani?

Secondo la teoria di Alan Turing (a cui si è accennato nel Capitolo 1) una macchina si ritiene "intelligente", e dunque più umana,[18] se in grado di imitare il comportamento (compreso quello linguistico) degli esseri umani. A un primo approccio, la risposta sarebbe, dunque, affermativa, eppure, come già detto, non si vogliono qui azzardare delle conclusioni inesatte che avrebbero, in realtà, bisogno di essere indagate molto più a fondo.

[17] L'idea che possa esistere una sorta di "copione" già fatto che la guida segue a ogni visita è, in un certo senso, confermata dall'ascolto parallelo delle visite guidate dalle quali si possono estrapolare categorie tematiche ricorrenti, come si avrà modo di vedere *infra* in 4.2.6.

[18] In questa sede si sta facendo la forse azzardata supposizione per cui *intelligenza* sia sinonimo di *umano*.

L'intenzione, ancora una volta, era semplicemente quella di dimostrare la ricchezza di riflessioni e di spunti che lavori su questa tipologia di dati inevitabilmente comportano.

4.2.2 *Inter-Annotator Agreement*

Prima di procedere ulteriormente nella descrizione della fasi che hanno portato alla messa in piedi del dataset di lingua parlata, ci si vuole qui un momento soffermare per parlare dell'importanza di avere una base di scientificità per l'annotazione delle visite.

Come si diceva in 4.1, la parte di annotazione è stata effettuata in parallelo con un altro tirocinante, Federico Boggia, e, in particolare, è stata da entrambi effettuata l'annotazione dell'intera visita 3 con lo scopo di voler controllare l'affidabilità dell'annotazione stessa e verificare che il processo di *sentence splitting*, seppur difficile da definire, potesse essere inconsciamente effettuato da due annotatori umani in maniera pressoché oggettiva. Il dubbio era quello, infatti, che la percezione di frase nel parlato fosse soggettiva e, di conseguenza, cambiasse di parlante in parlante.

Inoltre, uno dei propositi era quello che il mio studio si discostasse ulteriormente da quello di Boggia nella quantità dei dati processati, acquisendo i materiali della visita 1 da lui annotati; ma, per integrare questi ultimi con i dati in mio possesso, era necessario assicurarsi della similarità dell'annotazione risultante, per evitare di compromettere il dataset finale nella sua interezza.

Alla luce di tutto questo, era necessario procedere al calcolo dell'accordo tra i due annotatori. Il calcolo di quello che viene, in letteratura, definito come *Inter-Annotator Agreement* (IAA), è molto frequente negli studi linguistici proprio per valutare il grado di affidabilità di un'annotazione. Poiché esistono diverse metriche, di solito la scelta ricade su quella più adatta a seconda del task che si è effettuato e che, dunque, si vuole valutare.

Il task da noi eseguito, ad esempio, rientrava in una particolare categoria definita *unitizing* che raggruppa tutti quei task in cui il compito dell'annotatore è quello di identificare il confine di un determinato elemento linguistico: nel nostro caso, l'unità alla base della lingua parlata.[19]

In tal caso, il tipo di notazione da noi scelta per il calcolo dell'agreement è una rivisitazione dell'*IOB format* che viene, appunto, solitamente usato per marcare token che appartengono a una data entità da individuare (il caso più tipico da citare è quello di una *named-entity*).[20] L'idea è quella di marcare:

- Con I (*Inside*) i token che appartengono a una data entità;
- Con O (*Outside*) i token che non vi appartengono.
- Con B (*Beginning*) il token di apertura;

[19] Cfr. Gagliardi (2018).
[20] Per gli interessati, si rimanda al paper di riferimento Ramshaw e Marcus (1995).

A questi 3 tag previsti di default, se ne aggiunge un terzo, E (*End*), che, come si può ben immaginare, serve per marcare il token di chiusura dell'entità in questione.

Un ulteriore decisione da prendere riguardava quale trascrizione del testo mettere a confronto. La scelta ricadeva tra le tre a nostra disposizione:

1. La trascrizione ortofonica contenuta nel file eaf (comprendente i tag del progetto CLIPS[21] che segnalano tutti i fenomeni tipici del parlato come allungamenti vocalici, riempitivi, pause).

   Esempio 4.2.6. <inspiration> <vocal> <ehm> <tongue click> Allora abbiamo attraversato in particolare vi volevo segnalare il Chiostro dei Procuratori <sp>

2. La trascrizione fedele in termini di disfluenze, ma dalla quale sono stati eliminati tutti i tag di cui sopra si è detto.

   Esempio 4.2.7. Allora (abbiamo attraversato)+ in particolare vi volevo segnalare il Chiostro dei Procuratori.

3. La trascrizione ripulita di tutto fuorché del testo finale.

   Esempio 4.2.8. Allora in particolare vi volevo segnalare il Chiostro dei Procuratori.

Queste ultime due tipologie di trascrizioni erano disponibili nei file csv su cui si è lavorato durante le prime fasi di annotazione del dataset.

Quel che non si è detto, infatti, è che nella fase di segmentazione del discorso in frasi, si è utilizzata, come supporto per seguire le tracce audio, la trascrizione al punto 2 che, rispetto alla trascrizione ortofonica del file eaf (punto 1), non conteneva al suo interno i tag indicanti i fenomeni tipici del parlato.

Sulla trascrizione n° 2 si è proceduto, in questa fase, da un lato, ad annotare tutti i fenomeni di disfluenza (ovvero le ripetizioni di sillabe o gruppi di parole) inglobandoli all'interno di parentesi tonde seguite dal simbolo + e, dall'altro, a ripulire il testo da questi ultimi dando origine alla trascrizione "pulita" visibile nell'esempio 4.2.8.

La scelta più logica per confrontare il lavoro di annotazione svolto sarebbe stata quella di utilizzare quest'ultima trascrizione ripulita sia dai suoni extralinguistici che dalle disfluenze. Eppure, poiché era stato richiesto ai due annotatori di rendere il testo in una forma più leggibile introducendo i necessari segni di interpunzione, non vi era la garanzia che la trascrizione pulita dei due annotatori corrispondesse, in termini di token. Si sarebbe, cioè, dovuto procedere a un POS-tagging per poi eliminare i segni di punteggiatura, ma questo sembrava un processo eccessivamente lungo e macchinoso per l'obiettivo finale che si voleva raggiungere.

[21] Cfr. *supra*, par. 3.2.3.

A titolo esemplificativo, si faccia riferimento alla Tabella 4.1 in cui si mostra un disallineamento dovuto non solo alla punteggiatura ma anche a una diversa interpretazione, e dunque annotazione, del fenomeno delle disfluenze (l'annotatore 1, infatti, rimuove totalmente i token segnalati, in tabella, con un asterisco).

Annotatore 1	**Annotatore 2**
Però	Però
-	,
insomma	insomma
-	,
è	è
per	per
capire	capire
*	se
*	,
*	insomma
*	,
se	se
faccio	faccio
riferimento	riferimento
alla	alla
vita	vita
della	della
città	città
...	...

Tabella 4.1: Esempio di disallineamento tra i due annotatori nella trascrizione pulita. Il trattino indica il mancato inserimento della punteggiatura; l'asterisco l'espunzione di gruppi di parole da parte di un annotatore.

La scelta di non utilizzare la seconda tipologia di trascrizione era dovuta al fatto che il dataset finale fornito da Boggia era privo di quest'ultima: sin dalle primissime fasi preparatorie del dataset era stata, infatti, considerata una semplice "trascrizione di appoggio" da rimuovere a posteriori una volta ottenuto il testo ripulito.

L'unica base comune dalla quale i due annotatori partivano era, dunque, la trascrizione ortofonica dei file eaf.

L'unica problematicità da affrontare in questo caso, consisteva nella possibilità che, nella fase di *sentence splitting*, uno dei due annotatori avesse fatto terminare prima (o dopo) una frase rispetto all'altro, eliminando (o inglobando al suo interno) uno o più token[22] e dunque facendo convergere nella lista di token annotati con i tag IOB un numero differente di elementi, rendendo ap-

[22]In questo caso con il termine *token* si sta a indicare ogni singolo elemento contenuto in una riga del Grid di ELAN. Una definizione di token, dunque, più ampia che starebbe a inglobare anche i tag derivanti dal progetto CLIPS.

parentemente difficoltoso il processo di comparazione tra i token e gli annessi tag. In realtà, la notazione prescelta risolveva intrinsecamente il problema in quanto la soluzione prevedeva l'annotazione di tutti i token presenti all'interno della trascrizione ortofonica: eventuali token non rientranti nei confini di frase sono stati annotati come *outsider*, piuttosto che non essere affatto considerati.

Una volta ottenuta la lista di tutti i token, nessuno escluso, era possibile, quindi, procedere al calcolo dell'*agreement* facendo una comparazione token per token dei tag associati, come si può vedere dalla Tabella 4.2. In essa si osserva anche l'uso del tag O per segnalare i token esterni al segmento frasale individuato.

Trascrizione Eaf	**Tag Annotatore 1**	**Tag Annotatore 2**
<inspiration>	O	O
però	B	B
insomma	I	I
è	I	I
per	I	I
capire	I	I
se<eee>	I	I
insomma	I	I
<inspiration>	I	I
se	I	I
faccio	I	I
riferimento	I	I
alla	I	I
vita	I	I
della	I	I
città	I	I
...	...	...

Tabella 4.2: Esempio d'uso della notazione IOB.

Per effettuare il calcolo dell'*agreement* si è realizzato un piccolo script (*script_ calcolo_ agreement.py* che si riporta in allegato alla presente Tesi) che non fa altro che:

1. Processare il file con estensione eaf (mediante l'ausilio del pacchetto *pympi*)[23] estraendone il testo;

2. Assegnare a ogni token un tag (I, O, B o E) confrontando gli indici di inizio/fine parola presenti nella trascrizione allineata di ELAN con gli indici di inizio/fine frase individuati dai singoli annotatori e valutando, perciò, se il token in questione rientri o meno nel segmento individuato da ognuno di questi ultimi.[24] Al termine di questo step viene prodotto un file csv per ogni annotatore;

[23] Pagina di documentazione: `https://github.com/dopefishh/pympi`.
[24] Lo script annota le frasi composte da una sola parola con il tag E.

3. Comparare i risultati e calcolare l'*agreement*.

L'accordo tra gli annotatori viene calcolato in termini di accuratezza mediante la seguente formula:

$$agreement = nAgr * 100/token$$

dove:

- $nAgr$ = numero dei casi in cui i due annotatori si trovano d'accordo;
- $token$ = numero totale dei token annotati (che è uguale per entrambi gli annotatori).

I risultati mostrano valori di *agreement* soddisfacenti: 90,96% supera, difatti, di gran lunga il minimo valore percentuale accettabile (che di solito si aggira intorno al 70%).

Alla luce del valore di *agreement* ottenuto, è verosimile pensare che, nonostante i due annotatori non siano mai stati "addestrati" ad eseguire il task di *speech segmentation*, la conoscenza innata della loro lingua madre sembra essere sufficiente per individuare le unità alla base del parlato senza che vi siano sostanziali differenze nell'interpretazione del fenomeno oggetto di studio.

4.2.3 Estrazione delle feature linguistiche

Una volta in possesso dell'intero corpus di frasi, è stato possibile procedere all'annotazione delle feature linguistiche. Queste ultime sono state estratte automaticamente attraverso il tool *Profiling-UD*, di cui si offre una panoramica dettagliata in 4.2.3.1.

Nel preparare i file txt (rigorosamente in UTF-8), da dare in pasto allo strumento, si è proceduto a ricontrollare la pulizia del testo eliminando eventuali "rumori". Ad esempio, da segnalare è l'espunzione dell'ultima frase del POI 5 visita 3, in quanto troncata alla fine,[25] nonché l'eliminazione di tutte quelle frasi formate interamente da disfluenze.[26]

Tutti i file txt ottenuti (comprensivi delle visite 1, 2 e 3) sono stati compressi e dati in pasto al tool come unica zip.

Settando le opportune opzioni (lingua italiana, analisi delle singole frasi e testo presegmentato), si ottengono 3 file di output le cui caratteristiche verranno descritte più avanti.[27] L'unico file di interesse per la presente analisi è il secondo, ovvero quello contenente il profilo linguistico delle frasi del dataset.

Delle 129 feature estratte si è deciso di rimuoverne alcune. In particolare, *n_sentences* e *n_tokens* poiché, trattandosi di un'analisi per frasi, da una lato, il numero di queste era sempre costante (1) e, dall'altro, l'informazione contenuta nella colonna *n_token* era una replica di quella contenuta nella colonna *tokens_per_sent* e, pertanto, ridondante.

[25] Invece di concludersi con "il Tesoro", l'audio si interrompe a "(il Teso)+".
[26] Un esempio di quest'ultimo caso è presente nel file G01V02P06: "(Se se insomma s)+".
[27] Cfr. *infra*, sottosezione 4.2.3.1.

Una volta ottenuto il profilo linguistico delle frasi, esso è stato, in un primo momento, messo da parte per poi essere successivamente utilizzato per la costruzione del dataset finale come si vedrà nel paragrafo conclusivo 4.3.

4.2.3.1 *Profiling-UD*

Profiling-UD è un tool per l'analisi testuale, fortemente basato su *Universal Dependencies*,[28] e pensato, come rivela il suo stesso nome, per fare profilazione linguistica di un testo.[29]

Per *profilazione linguistica* si intende l'estrazione di un insieme di informazioni da un testo con l'obiettivo di classificarlo, o meglio, farlo rientrare in una data tipologia di appartenenza sulla base delle caratteristiche testuali rilevate. Una delle possibili applicazioni di questo task è, ad esempio, l'analisi stilometrica intesa come rilevazione dello stile autoriale di un testo al fine di poterne attribuire la paternità.

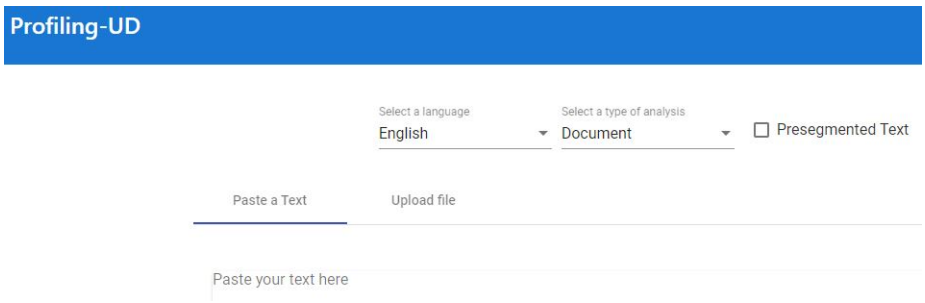


Figura 4.5: Schermata di input del tool *Profiling-UD*.

Il funzionamento di questo tool è estremamente semplice: un'interfaccia grafica molto intuitiva (vedi Figura 4.5) guida l'utente all'inserimento dei propri dati o come testo da incollare nell'opportuno riquadro, oppure come file da caricare. Inoltre, indispensabile risulta l'inserimento di una serie di informazioni utili al tool per effettuare correttamente l'annotazione.

In particolare, si richiede di specificare:

- La lingua di input;
- L'asse di analisi dei dati desiderato (se per documento o per singola frase);

[28] *Universal Dependencies* (UD) è un progetto che mira a dare delle linee guida per lo sviluppo di parser multilingui e fornire, conseguentemente, un inventario universale, valido per tutte le lingue, delle categorie per l'annotazione linguistica con eventuali estensioni specifiche per ogni lingua. La stretta dipendenza di *Profiling-UD* con UD è alla base della principale peculiarità di questo strumento: la possibilità di effettuare la profilazione per più lingue. Cfr. Brunato et al. (2020, p. 7145): «a main novelty of Profiling-UD is that it has been specifically devised to be multilingual since it is based on the Universal Dependencies framework» («la novità principale di Profiling-UD è che è stato specificamente concepito per essere multilingue dal momento che è basato sul framework di Universal Dependencies»).

[29] Il tool è disponibile al seguente link: `http://linguistic-profiling.italianlp.it/`.

- Presenza/assenza di una eventuale segmentazione del testo.

Uno step propedeutico alla profilazione linguistica di un testo è la sua annotazione[30] che viene eseguita da uno strumento integrato all'interno di *Profiling-UD*: *UDPipe* (Straka, Hajič e Straková 2016). Quest'ultimo effettua una serie di operazioni quali *sentence splitting* (se il testo fornito in input non è stato precedentemente segmentato)[31], tokenizzazione, *POS tagging*, lemmatizzazione e *parsing* delle dipendenze.

Come risultato di questo doppio step implementativo (annotazione e profilazione), il tool fornisce in output due diverse tipologie di file:

- Una zip contenente, per ogni documento di input, un file con estensione ConLLU[32] che rappresenta l'annotazione testuale.
- Un file csv, unico per tutti i documenti caricati, contenente il profilo linguistico, ovvero la totalità delle feature linguistiche estratte.

A questi file di output se ne aggiunge un terzo, in formato txt, che rappresenta l'elenco e la descrizione delle feature linguistiche estratte in fase di profilazione. *Profiling-UD*, infatti, non estrae sempre lo stesso numero di feature, bensì esso varia a seconda della tipologia di input che gli si fornisce.

Le feature linguistiche estraibili sono davvero moltissime in quanto coprono diversi livelli di analisi. Sarebbe impossibile fornirne un elenco completo, tuttavia, se ne fornisce una panoramica qui di seguito.

Come si legge in Brunato et al. (2020), esse sono raggruppabili in 7 macrocategorie:

1. Proprietà del testo grezzo (es. numero di frasi e token che compongono il documento, lunghezza media di ogni frase, lunghezza media di ogni parola)
2. Varietà lessicale (es. calcolando il *Type Token Ratio* o *TTR*[33]);

[30] Il termine *annotazione* è spesso usato in maniera molto generica per indicare una qualsiasi estrazione di informazione. In questo caso si parla dell'annotazione eseguita da uno specifico strumento che è *UDPipe*, che effettua operazioni molto mirate come *sentence splitting*, *tokenization*, estrazione di informazioni sintattiche, etc.

[31] Come si legge in Izre'el et al. (2020, p. 6), l'importanza di segmentare un testo deriva dalla possibilità di individuare, a posteriori, le relazioni morfosintattiche tra le parole. «when we have delimited our syntactic units in this way, we are ready to describe their make-up in terms of the word classes (parts of speech. . .) which appear in them» («una volta delimitate le unità sintattiche in questa maniera, possiamo descrivere il modo in cui in esse le parole si compongono a formare delle classi (parti del discorso...)»). Nel paper si parla, in realtà, di lingua parlata, ma questa riflessione si può applicare, nel caso di *Profiling-UD*, anche al testo scritto.

[32] Formato usato da *Universal Dependencies*. Si tratta di un formato contenente *plain text* codificato in UTF-8. Le righe che iniziano con il cancelletto riportano dei commenti e informazioni utili; quelle vuote marcano il confine tra le frasi; le rimanenti contengono, per ogni token, 10 informazioni separate da tabulazione. Per maggiori informazioni si consulti il sito web `https://universaldependencies.org/format.html`.

[33] Il TTR, che si calcola come il rapporto tra i tipi (ovvero il numero di occorrenze diverse degli elementi lessicali) e il numero di token totale nel testo, è un indice di ricchezza lessicale.

3. Informazioni morfosintattiche (es. informazioni relative alla distribuzione delle categorie morfosintattiche quali aggettivi, avverbi, nomi, pronomi...[34]);

4. Struttura dei predicati verbali (es. distribuzione delle teste verbali[35]);

5. Struttura sintattica a livello globale e locale (es. profondità media di un albero sintattico);

6. Relazioni sintattiche (es. distribuzione media delle 37 relazioni sintattiche usate nel progetto *Universal Dependencies*[36]);

7. Uso della subordinazione (es. numero di proposizioni subordinate e principali, posizione delle subordinate rispetto alle principali...).

In allegato alla presente Tesi è possibile consultare la legenda (*profiling-legend.txt*) fornita in output dallo stesso *Profiling-UD*, che costituisce una guida essenziale per la comprensione e interpretazione delle informazioni estratte.

4.2.4 Estrazione dei segmenti audio

Prima di procedere con l'aggiunta delle feature acustiche, era necessario essere in possesso dei segmenti audio per ogni frase del dataset.

A tal fine si è proceduto alla realizzazione di uno script in Python (*script_ segmentazione_ audio.py* visionabile in allegato) che, utilizzando un'apposita libreria (*Parselmouth*)[37], consente, appunto, di segmentare il file audio unico per la visita in tanti piccoli segmenti quanti il numero delle frasi contenute nel dataset finale ed estratte dalla corrispondente trascrizione testuale.

Il programma prende in input il dataset ripulito, però, da tutte quelle frasi contenenti esclusivamente disfluenze in maniera tale da ricavare solo i segmenti audio di frasi che avrebbero, di fatto, costituito il dataset finale. Come output il programma definisce, invece, per ogni visita, delle cartelle contenenti rispettivamente i segmenti audio estratti.

Una volta avvenuta la segmentazione, si è proceduto all'ascolto dell'inizio e della fine di ogni file audio prodotto per assicurarsi che le frasi, al loro interno, non risultassero troncate. Dopo aver apportato i dovuti aggiustamenti, si è nuovamente fatto rigirare lo script al fine di riestrarre i segmenti con gli start e gli end aggiornati.

In questa fase è doveroso segnalare l'impossibilità di poter definire, in ogni situazione, dei confini netti e precisi, nonostante l'utilizzo di ELAN consentisse di analizzare l'onda sonora dell'audio e, dunque, di essere estremamente scrupolosi nell'individuazione degli start ed end di frase. Spesso accadeva, infatti, che

[34] La lista completa delle 17 *part of speech* fondamentali si può trovare al seguente sito: `https://universaldependencies.org/u/pos/index.html`.

[35] Per *distribuzione delle teste verbali* si intende il numero medio di teste verbali in una frase paragonate al numero di proposizioni che occorrono in essa.

[36] Per la lista completa delle 37 relazioni sintattiche si rimanda a: `https://universaldependencies.org/u/dep/index.html`.

[37] Pagina di riferimento: `https://pypi.org/project/praat-parselmouth/`.

l'ultima parola di una frase fosse prosodicamente unita alla prima parola della successiva, rendendo difficoltosa l'individuazione del millisecondo preciso in cui si registrava il distacco tra una frase e l'altra.

Un esempio di ciò si ritrova in G01V03P02:

Esempio 4.2.9. Quindi avere dell'acqua buona, pulita, in realtà in alcune occasioni è stato molto più importante che semplicemente risolvere una questione pratica, cioè di comodità di avere dell'acqua pulita, pronta. In alcuni casi è stata proprio una questione di sopravvivenza

La *a* di "pronta" si lega fonicamente alla *i* di "in alcuni casi" della frase successiva rendendo ardua l'individuazione del punto esatto in cui termina la frase senza che vi sia l'impressione di una sua possibile prosecuzione.

La difficoltà di segmentare un discorso pensato per essere orale e che dovrebbe, pertanto, fluire ininterrotto è alla base anche della possibilità di trovare, tra gli audio estratti, alcuni che danno l'impressione che l'ultima parola sia troncata.[38]

Ad ogni modo, si è sempre cercato di individuare un punto di fine frase che potesse soddisfare l'orecchio di un eventuale ascoltatore umano.

Dei problemi si sono, inoltre, riscontrati con il file eaf G01V03PO6: a causa di alcuni disallineamenti consistenti tra l'audio e i token riportati nella trascrizione ortofonica di ELAN,[39] è stata necessaria una revisione manuale e una segmentazione più attenta del testo.

4.2.5 Estrazione delle feature acustiche

Estratti, dunque, i segmenti audio per ogni frase nella modalità descritta in 4.2.4, era possibile procedere all'estrazione delle feature acustiche caratterizzanti i singoli elementi del dataset.

Come di consueto, l'estrazione è avvenuta automaticamente mediante uno script allegato alla Tesi (*script_ estrazione_ feature_ acustiche.py*). In esso viene utilizzato *openSMILE* (Eyben, Wöllmer e Schuller 2010)[40], un toolkit (sviluppato in C++ ma di cui è disponibile un pacchetto per Python) utile proprio per l'estrazione di informazioni quali quelle di nostro interesse. L'obiettivo degli ideatori di *openSMILE* è quello di fornire agli utenti uno strumento semplice da usare, aperto a tutti e applicabile a una svariata molteplicità di campi (quindi non solo a studi che mirano a indagare le caratteristiche della lingua parlata, come nel nostro caso, ma anche, per esempio, a studi psicologico-comportamentali che debbano trattare suoni di qualsiasi tipo).

Nello script è definita una funzione che inizializza un oggetto settato in maniera tale da estrarre dal *feature set* disponibile (*ComParE_ 2016*) 65 feature di "basso livello" (e per questo definite *Low-Level Descriptor* o LLD).

[38] Ciò accade in diversi audio della visita 2.

[39] Per un esempio di trascrizione ortofonica si veda *supra*, par. 4.2.2.

[40] Il nome sta per *open-source Speech and Music Interpretation by Large-space Extraction*. Pagina di documentazione generica: `https://github.com/audeering/opensmile`. Pagina di documentazione specifica per Python: `https://github.com/audeering/opensmile-python`.

Gli LLD rappresentano un insieme di feature standard, selezionato, che viene adoperato in tutti quegli studi che vogliano utilizzare, per le loro analisi, diverse tipologie di informazioni. Queste ultime derivano, infatti, da svariati campi di ricerca quali quello dello *speech processing*, del *Music Information Retrieval* o semplicemente da settori che mirano ad analizzare in maniera generica i suoni.[41]

Poiché la comprensione del significato di ogni singolo descrittore richiederebbe delle conoscenze fisico-matematiche, i *Low-Level Descriptor*, estratti e utilizzati in questa sede, possono essere, piuttosto che descritti singolarmente, presentati a livello macroscopico secondo una categorizzazione fornita da Weninger et al. (2013) e visibile in Tabella 4.3.

Come si nota, gli assi di categorizzazione degli LLD sono due: da un lato i descrittori vengono raggruppati a seconda che si tratti di informazioni relative all'energia (*Energy related LLD*), allo spettro acustico (*Spectral LLD*) o al suono della voce (*Voicing related LLD*); dall'altro, ogni feature appartenente a uno dei 3 gruppi appena menzionati viene ulteriormente etichettata sulla base della tipologia di informazione trasmessa. In particolare si hanno: informazioni sulla prosodia (*prosodic*), sullo spettro o *ceptrum* acustico[42] (rispettivamente *spectral* o *cepstral*) e informazioni relative alla qualità del suono (*sound quality*).

	Group
4 ENERGY RELATED LLD	
Sum of auditory spectrum (loudness)	Prosodic
Sum of RASTA-style filtered auditory spectrum	Prosodic
RMS energy, zero-crossing rate	Prosodic
55 SPECTRAL LLD	
RASTA-style auditory spectrum, bands 1-26 (0-8 kHz)	Spectral
MFCC 1-14	Cepstral
Spectral energy 250-650 Hz, 1 k-4 kHz	Spectral
Spectral roll off point 0.25, 0.50, 0.75, 0.90	Spectral
Spectral flux, centroid, entropy, slope	Spectral
Psychoacoustic sharpness, harmonicity	Spectral
Spectral variance, skewness, kurtosis	Spectral
6 VOICING RELATED LLD	
F_0 (SHS and viterbi smoothing)	Prosodic
Prob. of voice	Sound quality
Log. HNR, Jitter (local, delta), Shimmer (local)	Sound quality

Tabella 4.3: Elenco dei 65 *Low-Level Descriptor* del ComParE feature set, raggruppati su due livelli a seconda della tipologia di informazione da essi veicolata. La tabella è stata presa dal lavoro di Weninger et al. (2013).

[41] Cfr. Weninger et al. (2013).

[42] Il *ceptrum* è un'informazione legata sempre allo spettro acustico, ma ottenuta a seguito di una trasformazione di quest'ultimo. Per tale ragione si può definire «lo spettro dello spettro» come si legge in *Cepstrum* (2014).

Come anche specificato in Weninger et al. (2013), per ottenere una visione più "soprasegmentale" e macroscopica degli LLD estratti, di solito si ha la tendenza a calcolare degli indici statistici, come quartili, percentili, media aritmetica, indici di dispersione e così via.

Seppure *openSMILE* fornisca già una versione funzionale degli LLD (un insieme definito, per l'appunto, *Functionals*), nel caso particolare di questo studio si è deciso di calcolare manualmente (mediante un apposito script) media, mediana e deviazione standard[43] di tutti valori estratti per ogni frame del segmento audio in questione. La ragione di ciò è dovuta al fatto che le trasformazioni effettuate dal tool sono spesso complesse e danno adito a un set di feature molto corposo (oltre 6 mila) che sarebbe stato sproporzionato rispetto al numero di dati di cui si poteva disporre.

I valori risultanti sono stati salvati in un apposito file di output che contiene, dunque, come risultato di questa operazione, 195 informazioni acustiche.[44]

Per riassumere quanto detto, le informazioni acustiche estratte in questo lavoro corrispondono a quelli che, in letteratura, vengono chiamati *Low-Level Descriptor*. Questi ultimi sono elencati (e raggruppati) in Tabella 4.3. Per ogni frase del dataset e per ogni descrittore, sono state calcolate media, mediana e deviazione standard, che ha avuto come risultato l'estrazione di ben 195 feature acustiche le quali sono state, infine, aggiunte al dataset finale.

4.2.6 Categorizzazione tematica delle frasi

Una volta definite le unità alla base della nostra analisi, misurato il grado di attendibilità dell'annotazione risultante ed estratte feature linguistiche e acustiche, si è proceduto all'aggiunta di ulteriori informazioni che, come si vedrà nei capitoli a seguire, avrebbero costituito le informazioni di partenza per la costruzione di due classificatori.

Un altro asse di variazione interessante riguardava il contesto tematico di appartenenza di ogni frase del dataset.

Dal momento che i nostri dati di partenza erano stati estratti da visite turistiche guidate, era ragionevole supporre che le informazioni veicolate dalla guida, di fronte a gruppi di visitatori differenti, fossero, in un certo qual modo, ricorrenti. Solitamente, infatti, durante un percorso guidato, una guida ha il compito di trasmettere ai visitatori una quantità minima (e quasi costante) di informazioni seguendo una sorta di "copione" già fatto. Era, cioè, lecito pensare di poter individuare delle categorie del discorso ricorrenti a cui ogni frase del dataset sarebbe appartenuta. Tale intuizione è stata, poi, confermata dall'effettiva facilità con cui è stato possibile individuare la ricorrenza di alcune informazioni nei discorsi fatti dalla guida ai vari gruppi in visita presso la Certosa di San Martino di Napoli.

[43] La scelta di calcolare anche la *standard deviation* è dovuta alla necessità di comprendere quanto i valori fossero sparsi rispetto a due indicatori di centralità quali media e mediana.

[44] Questo valore si ottiene moltiplicando il numero degli LLD (65) per il numero degli indici statistici selezionati (3).

L'aggiunta al dataset di un'informazione, per così dire, "tematica" avrebbe costituito, da una lato, il punto di partenza per la costruzione di un classificatore in grado di prevedere la categoria di appartenenza di una frase sulla base delle sue informazioni sintattiche e acustiche e, dall'altro, avrebbe aggiunto un'asse di variazione in più su cui il classificatore di *attention* avrebbe potuto basarsi.

L'annotazione delle categorie tematiche avrebbe rappresentato, inoltre, una novità assoluta rispetto a tutti i precedenti studi svolti dall'ILC meritatamente ai dati CHROME.

Una prima fase di rilettura delle trascrizioni audio ha, inizialmente, portato all'individuazione di una lista, piuttosto dettagliata, di tematiche comuni a ogni POI.[45]

A partire da queste liste, si è proceduto ad astrarre ulteriormente fino all'individuazione di sole 8 categorie che potessero, nel loro insieme, classificare la totalità delle frasi pronunciate dalla guida turistica nel corso delle varie visite.

Le categorie in questione sono:

- Categoria A - storia della Certosa: frasi in cui la guida fornisce informazioni sulla storia, costruzione e struttura del complesso monastico;

  Esempio 4.2.10. La Certosa di San Martino qui a Napoli ha almeno due anime.

  Esempio 4.2.11. Significa che questo luogo, in realtà a partire dagli anni Sessanta del 1800, è diventato anche museo.

  Esempio 4.2.12. Questi lavori di ammodernamento cominciano alla fine del 1500.

- Categoria B - informazioni storiche: frasi contenenti informazioni a carattere genericamente storico che non rientrano nelle categorie A (storia della Certosa) e C (informazioni biografiche). Si tratta di fatti ed eventi che si sono verificati in un luogo e/o in una data ben precisa;

  Esempio 4.2.13. Nel Trecento la collina del Vomero era pressoché disabitata, quindi si trattava di campagna.

  Esempio 4.2.14. Tra l'altro, Tesoro che venne fuso negli anni Novanta del 1700 per volere del re Ferdinando IV di Borbone, che in quel momento era lui ai ferri corti con i francesi.

[45] Giusto per avere un'idea del livello di dettaglio in cui si è scesi nella redazione di questa lista iniziale di argomenti, per il Pronao (*Point Of Interest* 1), ad esempio, sono state individuate macrotematiche come: presentazioni della guida e dei visitatori; spiegazione della struttura organizzativa del complesso monastico; riferimento alla recente trasformazione subita dalla collina del Vomero; riferimento alla storia della Certosa e alla dinastia angioina; informazioni relative all'architetto Tino di Camaino; riferimento ai lavori di costruzione e di ammodernamento della Certosa; indicazione degli archi a sesto acuto, della chiesa antica e del pavimento in marmi commessi realizzato da Cosimo Fanzago; indicazione delle cappelle e dei Santi cari all'ordine e così via.

Esempio 4.2.15. L'esempio più famoso, più eclatante, è quello della peste del 1656.

- Categoria C - informazioni biografiche: frasi che veicolano informazioni relative alla vita di personaggi storici menzionati quali regnanti, artisti, architetti e, nel caso di queste ultime figure, anche informazioni relative ai lavori artistico-architettonici da loro realizzati. Di solito, frasi appartenenti a questa categoria sono individuabili dalla presenza di una narrazione che assomiglia a quella che si leggerebbe nella sezione di un libro dedicata alla biografia di un personaggio realmente esistito;

  Esempio 4.2.16. Nella Napoli del Trecento sia Roberto che suo figlio facevano parte della dinastia angioina, quindi per una volta non gli spagnoli a Napoli ma i francesi.

  Esempio 4.2.17. Tra gli architetti che per primi hanno lavorato in questo luogo mi piace ricordare in particolare un senese, Tino di Camaino, che stava lavorando per il re Roberto.

  Esempio 4.2.18. Alcune sue opere le potete vedere in alcune delle maggiori chiese napoletane: Donna Regina, San Lorenzo, Santa Chiara.

- Categoria D - informazioni sui certosini: frasi che forniscono informazioni relative alla figura dei certosini e alla loro vita di clausura che coinvolge anche altre figure come i priori e i procuratori;

  Esempio 4.2.19. Certosini che passavano la maggior parte del loro tempo all'interno delle loro celle, per l'appunto, studiando, pregando, dedicandosi anche a lavori manuali, allo studio.

  Esempio 4.2.20. I procuratori, in particolare, si occupavano dei rapporti con il mondo esterno.

  Esempio 4.2.21. A controllare che tutto si svolgesse nel migliore dei modi era il priore.

- Categoria E - descrizione arte/architettura: frasi che sono volte a spiegare lo stile e/o la funzione di particolari ambienti o singoli elementi architettonici della Certosa, ma anche a descrivere le scene di affreschi e illustrare i significati simbolico-metaforici a esse collegati. In questa categoria rientrano, inoltre, frasi che sono volte a indicare e mostrare quello che è immediatamente visibile ai visitatori;

  Esempio 4.2.22. Forse, proprio sulla vostra testa, potete vedere quel che resta di alcuni archi a sesto acuto.

  Esempio 4.2.23. Tra l'altro qui, appunto, vedete Giuditta, con la testa di Oloferne, va dall'altra parte.

Esempio 4.2.24. E, oltre a questi, diciamo, grandi uomini e santi, al di sopra vedete dei certosini che si intervallano con delle figure femminili che sono delle allegorie di virtù.

- Categoria F - interazione: all'interno di questa categoria rientrano i saluti e le presentazioni della guida (che si ritrovano solo nei POI 1 di ogni visita) nonché quei segmenti che prevedono un coinvolgimento più o meno diretto del pubblico (e che di solito richiedono o possono richiedere un feedback da parte degli ascoltatori).

  Esempio 4.2.25. Io sono [nome], sono la storica dell'arte che questa mattina vi accompagnerà attraverso gli ambienti della Certosa di San Martino.[46]

  Esempio 4.2.26. Siete tutti napoletani?

  Esempio 4.2.27. Naturalmente la decorazione avviene in fasi, ok?

  All'interno di questa categoria rientrano, inoltre, anche tutte quelle risposte, spesso brevi, che la guida fornisce a seguito di una domanda di uno dei visitatori.

  Esempio 4.2.28. Sì.

  Esempio 4.2.29. Esatto.

  Esempio 4.2.30. No, no, proprio si chiudono praticamente dentro.

- Categoria G - meta-informazioni: frasi che contengono una sorta di recap di quello che si è visto, detto o fatto e di quello che si vedrà, dirà o farà;

  Esempio 4.2.31. Tra un attimo ci sposteremo alle spalle dell'altare per visitare il coro.

  Esempio 4.2.32. Più tardi, durante la nostra visita, avremo modo di vedere quello che c'è alle spalle dell'altare.

  Esempio 4.2.33. Allora dal coro ci siamo spostati e abbiamo attraversato la sacrestia.

- Categoria H - miscellanea: questa categoria ingloba tutti quei segmenti che non rientrano nelle categorie precedenti. Nonostante l'imprevedibilità dei segmenti del discorso che vi potrebbero ricadere, c'è una sorta di ricorrenza all'interno della categoria stessa. Ad esempio, vi si trovano spesso inviti al pubblico di vario tipo (come la frase riportata in 4.2.34) oppure informazioni di servizio relative ai lavori di manutenzione della Certosa.

[46] Come si può notare da questo esempio, un'altra delle manipolazioni fatte al dataset è la rimozione del nome della guida turistica al fine di anonimizzarlo.

Esempio 4.2.34. Quando ci sposteremo, vi invito insomma a guardarla perché vale la pena, soprattutto vi segnalo gli affreschi di Micco Spadaro.

Altri argomenti ricorrenti sono quelli dell'etimologia e della toponomastica oppure delle brevi parentesi che la guida apre per fare riflessioni e osservazioni personali.

Esempio 4.2.35. Io spesso e volentieri penso alle donne.

Spesso, in questa categoria, rientrano anche informazioni aggiuntive che la guida dice a seguito delle domande di un visitatore le quali, però, non rientrano nella categoria F (interazione) né in nessuna delle altre categorie. Per esempio, informazioni di carattere molto generico relative ad arte e cultura come nell'esempio 4.2.36.

Esempio 4.2.36. Quando fai un affresco, l'affresco poiché si fa direttamente sul muro, l'affresco è come se tendenzialmente fosse più chiaro.

Un prospetto sintetico e completo delle categorie è visibile in Tabella 4.4.

Categoria	Descrizione
A	Storia della Certosa
B	Informazioni storiche
C	Informazioni biografiche
D	Informazioni sui certosini
E	Descrizione arte/architettura
F	Interazione
G	Meta-informazioni
H	Miscellanea

Tabella 4.4: Le 8 categorie tematiche individuate.

La cosa interessante di questo processo di annotazione è che le categorie sono state definite considerando, inizialmente, solo le visite 2 e 3, ricevendo solo in un secondo momento i dati di Boggia relativi alla visita 1. Nonostante ciò, le categorie individuate sono state perfettamente in grado di raggruppare al loro interno le frasi di una visita che in principio non si conosceva, confermando, in un certo senso, il loro carattere di validità generale per qualsiasi discorso che segua il "copione" di cui sopra si parlava.

Certamente si è consapevoli del carattere arbitrario del processo di definizione e attribuzione delle categorie; eppure, come già sottolineato più volte, la facilità con cui questo step di annotazione è avvenuto sembrerebbe rendere un certo merito e una certa solidità alle categorie stesse.

4.2.6.1 Alcune linee guida

Per l'attribuzione delle categorie si sono seguiti, a grandi linee, dei criteri oltre che le varie definizioni date nel paragrafo precedente.

Innanzitutto, nel valutare il contenuto di ogni singola frase, si è cercato, il più possibile, di non farsi influenzare dal contesto immediatamente precedente di cui, per ovvi motivi, si era a conoscenza. Ovviamente si tratta di una regola priva del carattere di assolutezza che le dovrebbe appartenere: difatti, in certi casi, era impossibile non rifarsi al contesto antecedente. Sarebbe stato impossibile, ad esempio, far rientrare delle frasi come quelle riportate negli esempi 4.2.37 e 4.2.38 in una delle categorie sopra menzionate senza tenere in considerazione le informazione contestuali che le precedevano.[47]

Esempio 4.2.37. È Giuseppe Sammartino.

Esempio 4.2.38. Il famoso memento mori ricorda che devi morire.

Un'altra regola che si è cercato di rispettare è stata quella di assegnare, in presenza di una frase lunga con molte informazioni al suo interno, la categoria tematica sulla base del focus principale. Ad esempio, la frase riportata in 4.2.39 ricadrebbe apparentemente in E (descrizione arte/architettura); eppure, se si guarda più attentamente al focus del discorso, si noterà che l'informazione principale ruota attorno alla descrizione di un'abitudine propria della vita dei monaci in Certosa. Pertanto, l'intera frase è marcata come D (informazioni sui certosini).

Esempio 4.2.39. I certosini, che vivevano nelle celle che vedete attorno a noi, e le descriveremo tra un attimo, utilizzavano il chiostro anche come spazio di meditazione

Come anche nel caso del *sentence splitting*, sarebbe qui impossibile descrivere tutte le possibili casistiche incontrate e i vari ragionamenti fatti durante la fase di assegnazione delle categorie. Tuttavia, si spera che quanto detto possa rendere più chiara e trasparente possibile la procedura seguita in questo step di elaborazione del dataset.

4.2.7 Annotazione dell'attenzione

Come ampiamente spiegato nel capitolo 3, le informazioni relative all'*attention* derivavano da precedenti lavori di annotazione effettuati da Gomis e Poggianti su un sottoinsieme di dati CHROME.

L'annotazione, effettuata mediante il tool PAGAN, ha prodotto due file csv, uno per ogni annotatore, contenenti al loro interno 3 informazioni (vedi Figura 4.6):

- *Timestamp*, ovvero la data e l'ora dell'annotazione;
- *VideoTime*, il momento (in millisecondi) del video a cui si riferisce l'annotazione (che è anche il momento esatto in cui gli annotatori interagiscono con il tool, cioè premono una delle due frecce, in basso o in alto);

[47] La prima frase dell'esempio è stata, poi, difatti, marcata come E (descrizione arte/architettura) poiché rientra nell'ambito di una descrizione dell'ambiente circostante i visitatori; la seconda, invece, come D (informazioni sui certosini) in quanto, nel contesto precedente, si stavano dando delle informazioni circa gli argomenti di riflessione da parte dei monaci.

- *Value*, valore assegnato all'annotazione.

Timestamp	VideoTime	Value
1590240953567	0	0
1590240957554	3990	1
1590240957572	4008	2
1590240957588	4025	4
1590240957604	4041	5
1590240957621	4058	7

Figura 4.6: Come si presenta l'annotazione dell'attenzione fornita da Gomis.

A questo punto, era, dunque, necessario associare a ogni frase del dataset, di cui si disponevano gli start e gli end in millisecondi, un corrispondente valore di *attention* sulla base dei dati forniti, in particolare, da Gomis, ovvero l'annotatore il cui file presentava più valori e dunque meno possibilità che a una data frase del corpus corrispondessero zero valori.

Da sottolineare, infatti, che PAGAN tiene conto delle singole interazioni che l'annotatore ha con lo strumento di annotazione, pertanto è possibile avere dei *gap*: ciò significa, in sostanza, che è possibile non avere, per certi millisecondi del segmento audio, alcun valore assegnato. Altra cosa da tenere in considerazione è che il modo in cui l'*attention* è stata annotata tiene conto solo ed esclusivamente di un eventuale aumento o diminuzione del valore, senza registrare una persistenza di stato (si è constatato, infatti, attraverso un opportuno controllo, dell'assenza di valori di *attention* uguali, ripetuti uno dopo l'altro).

Pertanto, per ridimensionare i valori di *attention* contenuti nel file fornito da Gomis e traslarli nel dataset in mio possesso, era necessario andare a considerare, innanzitutto, la serie di valori registrati dall'annotatore che ricadevano entro l'intervallo di tempo definito dallo start e dall'end di una frase. Fatto questo, era sufficiente confrontare il valore di uscita al valore di ingresso: se maggiore, si sarebbe attribuita la classe 1 (aumento di attenzione); se minore, la classe -1 (diminuzione di attenzione); se uguale, la classe 0 (variazione assente). Nell'eventualità in cui non si registrava, per l'intervallo temporale in questione, alcun valore di *attention*, alla frase sarebbe stato associato il valore 99 (una classe appositamente introdotta per segnalare la casistica appena descritta).

La distribuzione ottenuta è mostrata in Tabella 4.5.

classe	**distribuzione**
99	213
-1	228
0	133
1	540

Tabella 4.5: Distribuzione delle classi: nessun valore (99), diminuzione di attenzione (-1), assenza di variazione(0), aumento di attenzione (1).

Sulla base del numero di esempi ottenuti per ogni classe e in virtù della similarità di significato tra le classi 99, 0 e -1, tutte indicanti, in un modo o

nell'altro, l'assenza di attenzione, è stato deciso di raggruppare queste ultime in maniera tale da renderle più equilibrate e confrontabili rispetto alla classe contrapposta, 1, indicante, invece, la presenza di *attention*.

Il risultato di questo raggruppamento è l'introduzione, nel dataset finale, di una nuova colonna (*engagement*) con, al suo interno, due valori possibili: 0 e 1, indicanti rispettivamente l'assenza o la presenza del fenomeno oggetto di interesse.

La distribuzione finale ottenuta, rispettivamente per ogni classe, è quella riportata in Tabella 4.6.

classe	**distribuzione**
0	574
1	540

Tabella 4.6: Distribuzione delle classi *engagement* (1) e *non-engagement* (0).

4.3 Il dataset finale

Alla luce delle manipolazioni fatte,[48] il dataset finale risulta composto da 1114 frasi (distribuite come mostrato in Tabella 4.7), ognuna delle quali descritta da 127 feature linguistiche, 195 feature acustiche, 1 informazione tematica e 1 informazione relativa al fenomeno di attenzione.

In totale il dataset contiene 328 colonne, in quanto, a quelle sopra menzionate, si aggiungono:

- La colonna *file*, che contiene un identificatore (univoco per ogni elemento del dataset) composto dal codice della visita e da un id numerico;
- Le colonne *start* ed *end* contenenti i millisecondi di inizio e fine di ogni frase;
- La colonna *text* che riporta la trascrizione "pulita"[49] della frase a cui si fa riferimento.

Per comporre questo dataset (dal quale, si ricordi, sono state eliminate alcune righe e colonne originariamente presenti)[50], si è proceduto ad allineare, sulla base della colonna *file*, gli output delle fasi di *sentence splitting*, l'estrazione di feature linguistiche e acustiche, la categorizzazione tematica e l'annotazione dell'*attention*.

Nel corso delle varie fasi di *pre-processing* del dataset si è proceduto, più volte, a ricontrollare le informazioni presenti al suo interno. Oltre ai controlli di "confine frase" di cui si è parlato in 4.2.4, numerosi check sono stati fatti in merito alla trascrizione del testo. Si è, ad esempio, controllata (e corretta)

[48] Cfr. *supra*, Figura 4.1.
[49] Cfr. *supra*, par. 4.2.2, esempio 4.2.8.
[50] Cfr. *supra*, par. 4.2.3.

Visita	Nome file	Ambiente	N° di frasi
1	G01V01P01	Pronao	82
	G01V01P02	Chiostro	55
	G01V01P03	Parlatorio	32
	G01V01P04	Sala del Capitolo	31
	G01V01P05	Coro	42
	G01V01P06	Stanza del Tesoro	74
2	G01V02P01	Pronao	68
	G01V02P02	Chiostro	61
	G01V02P03	Parlatorio	33
	G01V02P04	Sala del Capitolo	48
	G01V02P05	Coro	56
	G01V02P06	Stanza del Tesoro	79
3	G01V03P01	Pronao	117
	G01V03P02	Chiostro	84
	G01V03P03	Parlatorio	61
	G01V03P04	Sala del Capitolo	64
	G01V03P05	Coro	41
	G01V03P06	Stanza del Tesoro	86

Tabella 4.7: Struttura dei dati e distribuzione delle frasi per POI.

l'eventuale presenza di errori di battitura (es. **refertorio* invece che *refettorio*) ed errori ortografici (es. **perchè* al posto di *perché*). Inoltre, sono state adottate alcune convenzioni di scrittura, specialmente per le date: non **anni '60* ma *anni Sessanta*, non **Cinquantasei* ma *'56*, non **'300* ma *Trecento* e così via. Si è effettuata anche una normalizzazione di maiuscole e minuscole (es. *certosini* non **Certosini*, *Sala del Capitolo* e non **sala del capitolo*) e della punteggiatura.

A tal proposito, è doveroso ricordare che l'unico segno di interpunzione utilizzato per tenere conto dell'andamento intonativo con cui i gruppi di parole di una frase venivano pronunciati è stata la virgola.[51] Gli altri segni, infatti, come ad esempio il punto e virgola, avrebbero dato il senso di una pausa eccessivamente lunga difficilmente distinguibile, dal punto di vista intonativo, dalla pausa tipica di un punto. In tal senso, la loro introduzione avrebbe introdotto una difficoltà in più nel processo di divisione in frasi, che si è, ovviamente, voluta evitare.

Inoltre, una volta acquisiti da Boggia i dati della visita 1, nella fase di rilettura (necessaria per l'assegnazione delle categorie tematiche), si è proceduto a effettuare tutti i controlli di cui sopra si è appena parlato.

Il dataset è stato infine anonimizzato ricercando tutte le occorrenze del nome

[51] Per le convenzioni grafiche adottate nel corso dell'annotazione si rimanda *supra*, sottosezione 4.2.1.2.

della guida turistica e sostituendole con l'espressione "[nome]".[52]

Si tiene a precisare che la *pipeline* così com'è stata definita ha preso forma pian piano e non sono stati sporadici i casi di "retromarcia". Ad esempio, dopo aver estratto per la prima volta le feature acustiche, si è notata la presenza di valori molto sparsi (ovvero la ricorrenza di numerosi zeri). Ciò aveva, come causa, una corruzione dei file audio che ha reso indispensabile una nuova riestrazione delle feature in questione.

In questa sede, non si vuole procedere oltre nella spiegazione di ulteriori dettagli. Eppure, ciò che premeva, era dare consapevolezza del fatto che l'annotazione di un dataset è un processo estremamente lungo che richiede una grande attenzione e durante il quale è assolutamente lecito (e normale) commettere errori o fare dei ripensamenti andando a modificare ciò che, in precedenza, è stato fatto.

Ad ogni modo, questo dataset si prestava bene per l'addestramento di sistemi di classificazione automatici della cui natura si parlerà più in dettaglio nei prossimi due capitoli.

[52] Cfr. *supra*, esempio 4.2.25.

Capitolo 5

Accenni di *Machine Learning* e classificazione

Prima ancora di costruire i classificatori di cui si è accennato, è necessario dare una definizione formale di "classificazione" e comprendere il contesto entro cui un sistema di questo tipo nasce e si costruisce.

Il background teorico da cui attingere è quello del *Machine Learning*. Ovviamente, data la vastità della materia, non si avrà la pretesa di scendere troppo nel dettaglio ma, piuttosto, ci si limiterà a introdurre una serie di nozioni basilari utili alla comprensione del funzionamento macroscopico e generale del sistema.

5.1 Intelligenza Artificiale vs *Machine Learning*

Negli ultimi tempi, discipline come Intelligenza Artificiale (IA) e *Machine Learning* (ML) si sono affermate in moltissimi campi, migliorando e rendendo più semplice e sicura la nostra vita quotidiana sotto molti punti di vista.

In questa sede si parlerà solo ed esclusivamente di sistemi di apprendimento automatico che fanno capo a quest'ultimo settore di ricerca. Tuttavia, risulta importante comprendere almeno la relazione vigente tra l'IA e il ML che, contrariamente a ciò che possano pensare in molti, si pongono su un piano di stretta dipendenza: il *Machine Learning*, infatti, può ritenersi un'applicazione più specifica e ristretta di Intelligenza Artificale.

In entrambi i casi, l'obiettivo è quello di costruire sistemi "intelligenti" in grado di fornire una risposta adeguata a un input proveniente dall'ambiente esterno. La differenza risiede, però, nel paradigma di creazione di questi sistemi: se da un lato, lo sviluppatore dovrà occuparsi di prevedere (e programmare) il comportamento del sistema in qualsiasi scenario possibile, dall'altro ci si dovrà occupare unicamente di raccogliere (ed eventualmente preparare in maniera

opportuna) una certa quantità di dati[1] avendo cura di darli in pasto al sistema, che sarà, dunque, in grado di apprendere autonomamente il task desiderato.

Le protagoniste indiscusse del *Machine Learning* sono, in altre parole, macchine in grado di apprendere, come suggerisce, d'altronde, il nome della disciplina stessa.

Un approccio di questo tipo è sicuramente di aiuto in tutti quei casi in cui:

- Non si è in possesso di un modello teorico solido per formalizzare un dato fenomeno (in altre parole, quando non si è in grado di spiegare al sistema automatico esattamente come risolvere o svolgere il task assegnato);[2]
- Si possiedono dati rumorosi e/o incompleti.

In sostanza, il ML entra in gioco nel momento in cui bisogna trovare soluzioni a problemi difficili di cui sarebbe impossibile (o quasi) definire un algoritmo per la loro risoluzione.[3]

> «Cosa fareste se non riusciste a spiegare a un'altra persona la natura di un problema né i passi per arrivare alla sua soluzione? Probabilmente cerchereste alcuni esempi a partire dai quali il vostro interlocutore possa generalizzare per capire come comportarsi anche in situazioni in cui non si è mai trovato prima. Le tecniche di *machine learning* o *apprendimento automatico* agiscono proprio in questo modo. [...] Avendo a disposizione una grande quantità di esempi di soluzioni per il medesimo problema, il computer è in grado di operare una generalizzazione che gli permette di affrontare anche situazioni del tutto sconosciute» (Rossi 2019)

5.1.1 Alcuni esempi pratici

In generale, le applicazioni del ML sono vastissime e, al giorno d'oggi, siamo inconsapevolmente circondati da sistemi di questo tipo che, come accennato nel paragrafo precedente, rendono non solo più semplice ma spesso anche più sicura la nostra vita quotidiana.

Si pensi, ad esempio, ai meccanismi di classificazione delle email *spam*, che vengono indirizzate, senza un nostro particolare intervento, nella cartella di *Posta indesiderata*, diminuendo la probabilità di subire attacchi informatici.

Oppure si pensi anche ai meccanismi di riconoscimento facciale integrati in dispositivi elettronici di uso quotidiano (quali *tablet*, *laptop* e *smartphone*)

[1] È proprio l'enorme quantità di dati di cui sistemi di ML necessitano che rende spesso difficoltosa la messa a punto di queste macchine. Eppure, il processo di digitalizzazione, oggi, ha reso tutto molto più semplice in quanto si possono trovare moltissimi dati in formato elettronico e, dunque, già pronti per l'uso.

[2] Un esempio di ciò è proprio il processo di *speech segmentation* di cui si parlava *supra*, par. 4.2.1: come si potrebbe insegnare a una macchina a suddividere in frasi un segmento della lingua parlata? Data la difficoltà nel formalizzare una soluzione, l'unico modo sarebbe quello che la macchina impari da sé, al pari di quanto fanno i parlanti nativi di una lingua.

[3] Per algoritmo si intende, tradizionalmente e genericamente, una «qualsiasi procedura di calcolo per ottenere un qualche risultato». Nel campo più specificamente informatico e dell'IA, un algoritmo è una « sequenza di istruzioni che dicono come risolvere un problema», producendo un output a partire da un insieme di dati forniti in ingresso (Rossi 2019).

o installati nelle aree di controllo passaporti degli aeroporti, che rendono più rapido il processo di identificazione dell'identità delle persone.

Un'altra applicazione decisamente interessante è nell'ambito medico di cui si può fornire un esempio piuttosto recente. Lo scorso anno, infatti, in concomitanza con la diffusione della sindrome respiratoria acuta grave SARS-Cov-2, si è adottato, presso il Policlinico universitario di Roma, un sistema in grado di prevedere, a partire da immagini tomografiche dei polmoni, la possibile insorgenza dell'infezione polmonare nonché il suo possibile decorso fornendo un valido supporto a medici e infermieri del centro.[4]

5.2 Nozioni basilari di ML

Il *Machine Learning* è un campo tanto affascinante quanto complesso, a cui ci si può, in verità, approcciare in vari modi. Per comprenderlo a fondo sarebbero necessarie conoscenze in svariati campi come quello logico-matematico, filosofico, psicologico e statistico. Tuttavia, un sistema di classificazione, quale quello di nostro interesse, può essere spiegato in maniera macroscopica (ma non per questo superficiale), definendo e fissando una serie di concetti.

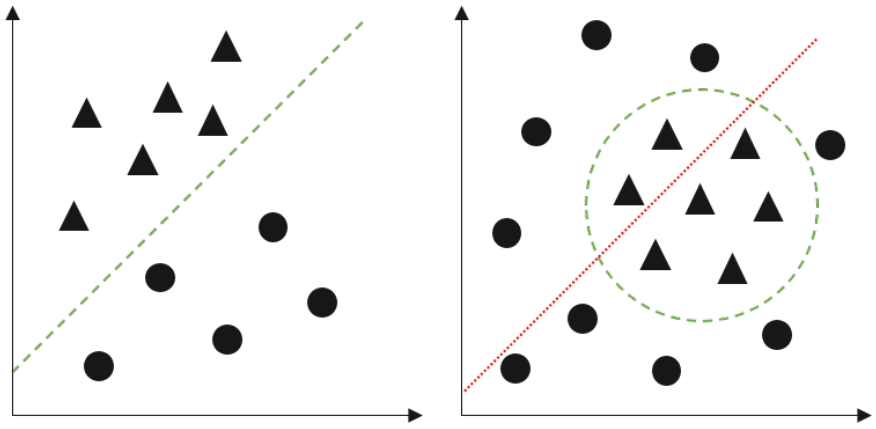
Il primo nodo da sciogliere è il seguente: come fa il sistema ad acquisire una conoscenza utile alla risoluzione di un determinato task? Tutto si basa su un semplice meccanismo: il sistema osserva dei dati che definiscono o rappresentano un certo fenomeno di interesse e, a partire dai dati, costruisce un modello teorico da applicare in situazioni simili.

Il concetto di *apprendimento*, dunque, formalmente, altro non è che un processo di ricerca e di approssimazione di una funzione matematica che potrà essere applicata dal sistema ogni qual volta gli si richieda di eseguire uno specifico compito. La funzione rappresenta, dunque, la conoscenza che il sistema ha di un determinato fenomeno e sulla base della quale effettua scelte o fornisce previsioni. Quale sia questa funzione (o quale sia la sua natura) poco importa, in quanto uno dei vantaggi dei sistemi di ML è proprio la possibilità di poterli utilizzare come delle *black box*.

In generale, però, si tenga presente che, a seconda della natura del modello (e dunque della tipologia di funzione che esso è in grado di trovare), dipenderà la capacità dello stesso di risolvere in maniera efficace, o meno, un problema. Ad esempio, la Figura 5.1 mostra le soluzioni che un modello lineare troverebbe per la separazione di triangoli da cerchi: in entrambi i casi si tratterebbe di una linea retta. Se nel primo scenario (Figura 5.1a) la soluzione ottimale coincide con la retta che il modello è in grado di trovare, nel secondo scenario (Figura 5.1b) ciò non accade. In quest'ultimo caso servirebbe, infatti, un modello non lineare per risolvere efficacemente il problema rappresentato.

Un'altra nozione importante nel campo del *Machine Learning* è la distinzione tra *supervised* e *unsupervised learning*. La differenza tra le due modalità di apprendimento risiede nella natura dei dati forniti al sistema durante la fase di addestramento e, di conseguenza, anche nella tipologia di task che saranno in

[4] Cfr. Altimari (2020).



(a) Problema linearmente separabile. (b) Problema non linearmente separabile.

Figura 5.1: Le figure mostrano rispettivamente un esempio di problema linearmente e non linearmente separabile. Le linee tratteggiate in verde rappresentano le funzioni che un buon modello troverebbe, rispettivamente, nelle due situazioni per il discernimento tra triangoli e cerchi. In rosso, invece, la soluzione (non ottimale) che un modello lineare troverebbe nel caso di un problema non linearmente separabile.

grado di eseguire: in un caso, si tratta di *labeled data*, cioè di dati etichettati con valori (numerici o categorici)[5] che il modello dovrebbe essere in grado di prevedere una volta conclusa la fase di addestramento; nell'altro caso, si tratta, invece, di *unlabeled data*. Qui il target è l'individuazione di pattern ricorrenti che permettono una maggiore comprensione dei dati forniti in ingresso.

Un contesto di applicazione degli algoritmi di apprendimento non supervisionato è il *data mining*, un campo di ricerca in forte crescita il cui scopo primario è quello di trasformare dati in conoscenza atta a fare delle previsioni. Ad esempio, nel settore del marketing, risulta, oramai, fondamentale riuscire a raggruppare i potenziali clienti in cluster di utenti simili in quanto questo consente di migliorare e affinare le tecniche di profilazione, *re-targeting* e, dunque, vendita dei prodotti.

Indipendentemente dal fatto che che si tratti di *supervised* o *unsupervised learning*, vi è sempre una fase di addestramento dell'algoritmo, che prende il nome di *training phase* durante la quale viene fornita al sistema una parte dei dati di cui si dispone. Questi andranno a costituire quello che, in gergo tecnico, è chiamato *training set*. Al termine di questa fase, l'algoritmo apprenderà una funzione, un modello, che, come già detto, rappresenta l'unica fonte di conoscenza che il sistema ha di un certo fenomeno.

[5]Per una maggiore precisione, se il task prevede l'attribuzione di un'etichetta numerica reale si parla di *regressione*, nel caso in cui si tratti, invece, di un valore che sia numerico discreto o categorico allora si parla di *classificazione*.

Nel caso specifico in cui si tratti di un algoritmo supervisionato è utile, inoltre, valutare la performance del modello testandolo su un set di dati nuovo (di cui, però, si conoscono gli output), che prende il nome di *test set*.[6] Poiché è proprio su questo che si andrà a valutare la capacità di generalizzazione del modello, è di fondamentale importanza che il sistema non abbia mai visto, durante la fase di training, i dati in esso contenuti, altrimenti il meccanismo di valutazione risulterebbe falsificato. Durante la fase cosiddetta di *evaluation*, infatti, non si fa altro che confrontare le etichette reali (che si conoscono a priori) con quelle predette (attribuite dall'algoritmo), ma se quest'ultimo ha già in precedenza visto quei casi, non farà altro che "barare" attribuendo un'etichetta precedentemente appresa.

L'importanza di valutare un modello scaturisce dalla presa di consapevolezza per cui i sistemi di *Machine Learning* non possiedono una conoscenza assoluta e onnicomprensiva del mondo, ma, al contrario, cercano di affrontare situazioni a loro ignote ricorrendo a tecniche statistiche che, per definizione, sono probabilistiche e, pertanto, incerte.

Da precisare, inoltre, che una pratica molto diffusa nel ML è quella di validare il sistema prima di passare alla fase di testing vera e propria.

Con *validazione* si intende una specifica fase del processo di costruzione di un modello di apprendimento durante la quale lo scopo primario è quello di fare *model selection*, ovvero di individuare e selezionare il modello migliore da portare all'effettiva fase di valutazione (anche detta di *model assessment*). Passare attraverso la fase di validazione è essenziale in tutti i quei casi si voglia fare, ad esempio, *parameter-tuning*, quando si vuole, in altre parole, trovare il settaggio più opportuno degli iperparametri del modello. La validazione rientra, dunque, in una fase ancora preparatoria e di definizione del modello rispetto alla fasi finali di testing e valutazione che prevedono la restituzione della stima del comportamento che si presume il sistema avrà al momento del suo utilizzo.

Un'ultima questione che qui si vuole affrontare relativamente al ML fa riferimento alla quantità dei dati da fornire al sistema durante la fase di addestramento. In linea generale, vale la regola per cui maggiore è la quantità dei dati, maggiore sarà la conoscenza acquisibile dal sistema e, dunque, la sua capacità di generalizzare e di gestire eventuali anomalie nei dati.

Banalmente, si pensi alle modalità di apprendimento degli esseri umani: riprendendo quanto detto nel paragrago 5.1, se dovessimo spiegare a una persona la natura di un problema, fornendo a quest'ultima il maggior numero di esempi possibile, le daremo una base più solida dalla quale partire per poter apprendere e, pertanto, generalizzare.

Questa regola, pero, non è solo dettata da buon senso o confermata dai fatti

[6] Nel caso di un algoritmo di apprendimento non supervisionato, valutare le performance del modello sarebbe superfluo in quanto si tratta di tecniche il cui scopo è l'esplorazione di dati al fine di estrapolarne conoscenza. In altre parole, non c'è un concetto di "giusto" o "sbagliato" in quanto tutto può essere conoscenza se viene correttamente interpretato. Non a caso, il compito di analizzare gli output di algoritmi non supervisionati è, di solito, affidato ad esperti specifici che prendono il nome di *data analyst*.

(in quanto proprio i cosiddetti *Big Data*[7] sono stati i responsabili di un netto miglioramento delle performance di sistemi di apprendimento automatico, specie nel campo dell'*image* e *speech recognition* e della traduzione automatica). Al contrario, grazie alla *Statistical Learning Theory*, è matematicamente dimostrabile che all'aumentare del numero di dati diminuisce il rischio che il sistema commetterà errori durante un suo possibile utilizzo.[8]

Per concludere, si vuole sottolineare che, in questa sede si parlerà solo ed esclusivamente del task di classificazione che si presenta come un problema di attribuzione di un'etichetta y (detta *label*) a un elemento x dotato di un insieme X ("X grande") di feature caratterizzanti.[9] Nei casi in cui il numero di label predefinite associabili a un elemento è pari a 2, si parla di *classificazione binaria*; in tutti gli altri casi, invece, in cui il numero di label possibili è superiore a 2, si parla di *classificazione multi-label*.

5.3 I modelli di classificazione

In virtù di quanto detto nel paragrafo precedente, per costruire un modello di ML e, in particolare, un modello di classificazione è necessario, perlomeno, passare attraverso le seguenti fasi:

1. *Training phase*;
2. *Testing phase*;
3. *Evaluation*.[10]

Si è anche detto che i sistemi di *Machine Learning* possono essere (e vengono spesso) utilizzati come delle scatole nere, senza curarsi del loro funzionamento interno. Nel caso del presente studio, in cui si voleva, da un lato, determinare quali feature prosodico-acustiche caratterizzassero il fenomeno di *attention* e, dall'altro, vedere se esistessero delle feature significative nel processo di attribuzione di una categoria tematica a una frase della lingua parlata, era necessario, però, usare degli algoritmi spiegabili, che consentissero, cioè, di comprendere i criteri attraverso cui il sistema fosse in grado di valutare. Per tale ragione la scelta è ricaduta su due semplici, ma al contempo potenti, algoritmi di apprendimento supervisionato di cui si fornirà una panoramica nei sottoparagrafi seguenti.

[7] Per l'affermazione dei *Big Data* ha certamente contribuito la diffusione di Internet e del *World Wide Web*: due eventi che hanno reso disponibili enormi quantità di dati digitalizzati e già pronti per essere processati da sistemi di apprendimento automatico.

[8] La SLT è una teoria, una legge universale, valida per qualsiasi modello di ML, che permette di inquadrare formalmente il problema di generalizzazione nonché il problema di *under* o *overfitting* che caratterizza questa tipologia di sistemi. L'aumento dei dati di training compensano anche un eventuale eccessivo aumento della complessità del sistema. Per un maggiore approfondimento in materia si rimanda a Bousquet, Boucheron e Lugosi (2004).

[9] L'insieme X rappresenta il *feature set* di un elemento.

[10] Sulla base di quanto detto nel paragrafo precedente, la fase di validazione è opzionale e richiesta solo nel caso di particolari esigenze sperimentali. Ad esempio, quando vi è la necessità di fare il *tuning* dei parametri.

Avere due modelli da addestrare (di cui analizzare performance e funzionamento) avrebbe aumentato la probabilità di individuare un modello efficace per la definizione dei fenomeni di interesse. Inoltre, per approfondire ulteriormente la conoscenza di questi ultimi, sono stati portati avanti diversi esperimenti secondo le modalità che si descriveranno nei capitoli 6 e 7.

Per la messa in piedi di entrambi i sistemi di classificazione si è deciso di utilizzare una libreria *open-source* di Python: *scikit-learn* (o *sklearn*), che costituisce un valido strumento per la costruzione di modelli per l'apprendimento automatico.[11]

L'importanza di questa libreria risiede nella possibilità di disporre di una serie di funzionalità, una per ogni step di cui prima si parlava. Di nostro particolare interesse sono state, infatti:

1. La funzione *fit*, responsabile della fase di addestramento del modello;
2. La funzione *predict*, responsabile della fase di testing (intesa come effettivo utilizzo del modello su dati nuovi);
3. La funzione *classification_report* (contenuta nel pacchetto *metrics*), che permette la visualizzazione di tutta una serie di metriche utili per la valutazione del modello.

5.3.1 SVM e SVC

La *Support Vector Machine* è un algoritmo di apprendimento supervisionato che oggi costituisce uno dei metodi di predizione più robusti per eseguire task di classificazione, regressione o *outlier detection*.[12] Alla base della sua affermazione vi è un meccanismo semplice, ma al contempo elegante e solitamente performante.

L'apprendimento avviene mediante l'osservazione dei dati forniti durante la fase di addestramento ed etichettati con l'opportuno output. Questi ultimi vengono proiettati, sotto forma di punti, in uno spazio n-dimensionale a seconda del numero di feature descrittive dei dati stessi. Lo scopo dell'algoritmo è quello di trovare un iperpiano di *n-1* dimensioni in grado di separare i punti appartenenti alle differenti categorie.

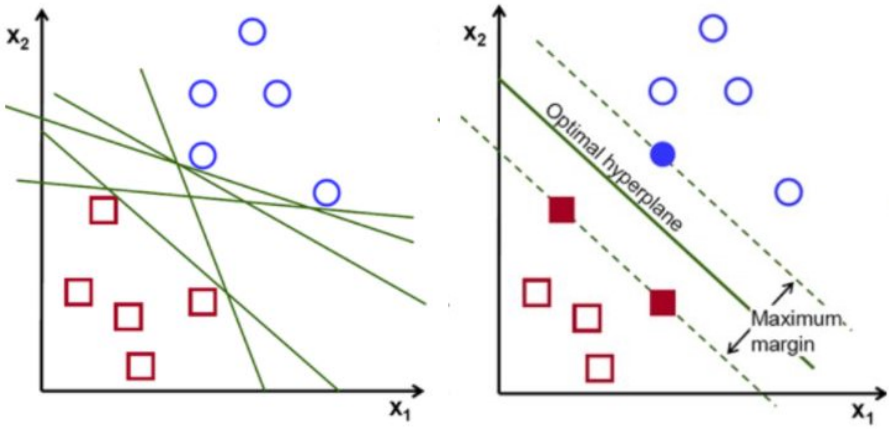
Uno dei grandi vantaggi di SVM è che il modello non è solo in grado di trovare *una* soluzione, ma di trovare *la* soluzione ottimale: se esiste, infatti, più di un iperpiano in grado di separare i valori di una classe dagli altri, la *Support Vector Machine* selezionerà l'iperpiano che massimizza il *margine*. Quest'ultimo si definisce come la distanza tra i punti più vicini all'iperpiano separatore, a loro volta definiti come *vettori supporto* o *support vector* (da qui, non a caso, il nome dell'algoritmo).[13] La Figura 5.2 dovrebbe chiarificare quanto appena esposto:

[11] Pagina di documentazione ufficiale: `https://scikit-learn.org/stable/`.

[12] Per *outlier detection* si intende l'individuazione di valori anomali all'interno di un set di dati.

[13] Da quanto detto, si intuirà che i vettori supporto sono gli unici punti dello spazio che contribuiscono alla definizione dell'iperpiano separatore, nonché della soluzione che SVM of-

in 5.2a si nota che per separare i quadrati dai cerchi esistono diversi possibili iperpiani; in 5.2b è raffigurato, invece, l'iperpiano che massimizza il margine e che verrà, perciò, trovato da SVM. In quest'ultima Figura i punti più vicini all'iperpiano (che ricadono lungo le righe tratteggiate) sono i vettori supporto.



(a) (b)

Figura 5.2: (a): visualizzazione dei possibili iperpiani separatori. (b): selezione dell'iperpiano ottimale. Entrambe le immagini sono state prese da Ippolito (2019).

Poiché, però, l'SVM è progettata per trovare, come soluzione a un problema di classificazione, un iperpiano (che per quante dimensioni abbia rimane comunque un prodotto scalare e, dunque, lineare), potrebbe risultare complesso risolvere un problema nel caso in cui quest'ultimo non fosse linearmente separabile (cfr. Figura 5.1). Anche questo ostacolo viene, in realtà, superato grazie al cosiddetto *kernel trick*: alla possibilità, cioè, di specificare un *kernel*, ovverosia una funzione matematica che SVM utilizzerà per traslare i dati in altre dimensioni fintanto che non troverà l'iperpiano separatore. Ciò consente al modello di rimanere lineare, ma, al contempo, risolvere problemi non linearmente separabili. La Figura 5.3 dà un'immagine molto chiara di cosa si intenda per *kernel trick*: se in uno spazio bidimensionale i punti non sono linearmente separabili, a seguito di una trasformazione degli stessi in uno spazio n-dimensionale, potrebbe essere possibile trovare l'iperpiano separatore desiderato.

All'interno del pacchetto *sklearn*, SVM dispone di diverse classi capaci di eseguire un task di classificazione. Tra queste vi è SVC (*Support Vector Clas-*

frirà: muovendo, in altre parole, gli altri punti/vettori la soluzione rimarrà invariata. Ciò costituisce un enorme vantaggio anche in termini di memoria in quanto, a essere memorizzati dal modello, non saranno tutti i punti dello spazio ma un sottoinsieme di esso: per l'appunto, i vettori supporto.

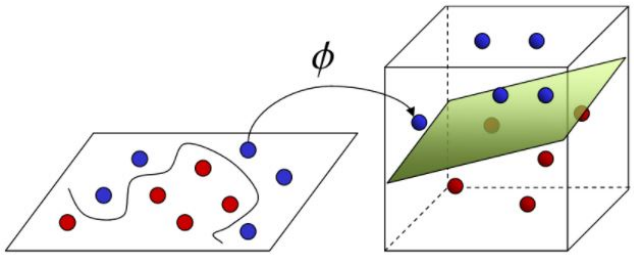


Figura 5.3: *Kernel trick*. L'immagine è stata presa da Ippolito (2019).

sifier)[14] che, in questo studio, è stata utilizzata, per entrambi i classificatori, con i parametri di default, specificando solo, come tipo di kernel, quello lineare. Tale scelta deriva dalla necessità di voler utilizzare un attributo di SVC, *coef_*, disponibile solo in quest'ultimo caso. L'attributo in questione avrebbe dato la possibilità di analizzare i pesi assegnati a ciascuna feature e di studiare e indagare più approfonditamente i fenomeni a cui il classificatore avrebbe fatto riferimento.

```
# model definition
clf = svm.SVC(kernel="linear")

# training phase
clf.fit(x_train, y_train)

# testing phase
y_pred = clf.predict(x_test)

# evaluation
report = classification_report(labels_true, labels_predicted,
    target_names = ['engagement', 'no_engagement'])
```

Figura 5.4: Fase di definizione, addestramento, test e valutazione del modello SVM.

Poiché *sklearn* fornisce le funzionalità di cui si è parlato in 5.3, l'implementazione dell'algoritmo avviene, di fatto, in poche righe di codice che si riportano in Figura 5.4: con *x_train* si definisce l'insieme di dati non etichettati passati al classificatore durante la fase di addestramento, mentre con *y_train* le rispettive label; *X_test* rappresenta, invece, un insieme nuovo (mai visto) di dati non etichettati conservati appositamente per la fase di testing; infine, *labels_true* e *labels_predicted* rappresentano rispettivamente le vere etichette e quelle predette dal classificatore dei dati forniti durante la fase di testing. Il codice mostrato nella Figura è stato estrapolato dallo script *script_classificatore_11FoldCrossValidation.py* (visionabile, nella sua interezza, in allegato alla Tesi), ma vi è stata

[14] Questa implementazione si basa sulla libreria *libsvm*, per cui si rimanda a Chang e Lin (2011).

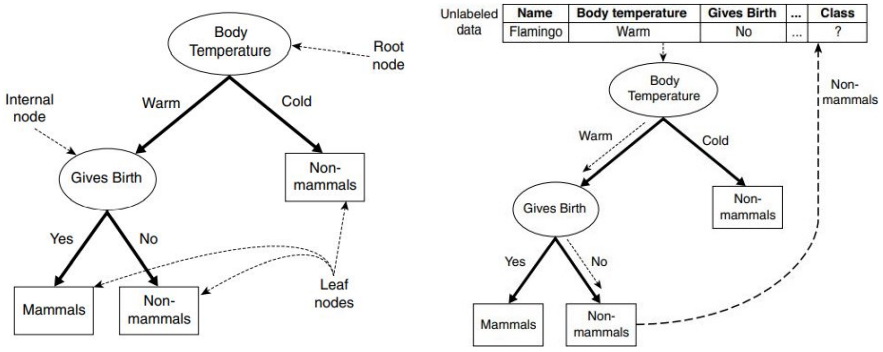
apportata qualche piccola modifica, in termini di nomenclatura delle variabili, al fine di renderlo più generico.

5.3.2 *Decision Tree* e *Random Forest*

Al pari di SVM, anche gli *alberi decisionali* (DT da *Decision Tree*) sono algoritmi di apprendimento supervisionato utili sia per task di classificazione che di regressione e dal funzionamento estremamente semplice.

Nel caso della classificazione, si tratta, come sempre, del problema di attribuire una label a un dato elemento di un sistema sulla base di alcune variabili che lo caratterizzano. Come avviene, però, esattamente il processo decisionale? L'algoritmo non fa altro che costruire un vero e proprio "albero", un grafo, cioè, caratterizzato da n nodi (*decision nodes*), ognuno dei quali corrisponde a una dato attributo del dataset. Da ognuno dei nodi (a meno che non si tratti di nodi foglia terminali) si dipartono, poi, n rami: 2 soli se si tratta di un *albero binario*, o più di 2 (rispettivamente uno per ogni valore possibile dell'attributo di cui il nodo è rappresentante), se si tratta, invece, di *alberi non binari*.

Il processo di decisione che consiste nell'applicare iterativamente, a ogni nodo, delle regole relative al valore assunto dalla feature per quello specifico elemento, si interrompe al raggiungimento di un nodo foglia, responsabile della predizione.



(a) Gli elementi costitutivi di un DT. (b) Classificazione di un nuovo elemento.

Figura 5.5: (a): elementi costitutivi di un albero decisionale. (b) descrizione del processo decisionale di un DT. Entrambe le figure sono state tratte da Tan, Steinbach e Kumar (2014).

La Figura 5.5a mostra gli elementi costitutivi di un albero decisionale: il nodo radice, i nodi interni (in cui avvengono delle decisioni che determinano, a loro volta, la direzione da seguire) e i nodi foglia, che contengono, invece, le label da assegnare. In Figura 5.5b si vede, invece, il percorso decisionale (tratteggiato) seguito dall'algoritmo nel processo di assegnazione della label ad un nuovo (e non etichettato) elemento fornitogli in input. Nell'esempio ci si chiede quale sia la temperatura corporea di un fenicottero e se sia in grado di dare alla luce un

cucciolo. Le relative risposte portano a un nodo foglia contenente l'etichetta "non-mammifero" che sarà, dunque, assegnata al record in questione.

Non si vuole qui approfondire l'aspetto di come avvenga, in sé, la fase di costruzione dell'albero (che varia, tra l'altro, a seconda dello specifico algoritmo prescelto), ma basti sapere che il processo di definizione dei nodi avviene, genericamente, sulla base di una serie di metriche utili a misurare il grado di purezza del nodo, ovvero sulla base della sua capacità di separare gli esempi in due o più gruppi tra di loro omogenei.[15] Per far questo sono necessarie delle misure di purezza o, viceversa, di impurità del nodo. Nel caso di *sklearn* le funzioni disponibili per misurare la qualità di un nodo sono il *Gini index* e l'entropia.

Per superare alcuni dei limiti intrinseci degli alberi decisionali, per cui, ad esempio, un albero è estremamente sensibile ai *training data*, sono stati introdotti nuovi algoritmi originatesi proprio dai DT. Tra questi i cosiddetti *Random Forest*.

Questi ultimi altro non sono che un insieme di alberi decisionali (da cui il nome *forest*) addestrati su sottoinsiemi casuali di dati estratti dal *training set* originario sia in termini di righe che di colonne. Ciò li rende molto più robusti e meno vincolati ai dati di training di quanto lo sia, invece, un singolo albero decisionale.[16] Se durante la fase di addestramento si andranno a definire un insieme di alberi decisionali, durante la fase di testing, invece, si andrà a consultare la decisione di ogni singolo albero e ad assegnare, al nuovo record, la label fornita come output dalla maggior parte degli alberi contenuti nella *forest* stessa.

Nella libreria *sklearn* l'algoritmo in questione è contenuto all'interno del modulo *ensamble*. Anche in questo caso, i parametri della funzione sono stati mantenuti ai loro valori di default e, come nel caso di SVM, le righe di codice (ovverosia le operazioni che, di fatto, contribuiscono alla definizione del modello) sono poche e del tutto equivalenti a quelle mostrate in Figura 5.4, con l'unica modifica della riga 2 che si presenterebbe come di seguito:

$$clf = RandomForestClassifier()$$

5.4 Addestramento, test e valutazione del modello

Indipendentemente dal tipo di classificatore che si seleziona, la prima cosa da fare è suddividere il dataset iniziale in x set, ognuno corrispondente a ciascuno degli step che si vuole eseguire. Per fare un esempio, se si vorrà procedere, come nel nostro caso, alle sole fasi di training e test, si renderà necessario essere in possesso di 2 set distinti, rispettivamente il training e il test set, ognuno con

[15] Per omogeneo si intende un gruppo contenente elementi afferenti a una sola classe.

[16] L'idea alla base degli alberi decisionali è la cosiddetta *wisdom of the crowd* (la "saggezza della folla"). L'idea, cioè, per cui il giudizio di un gruppo di individui è solitamente più attendibile e meno *biased* rispetto a quello di uno solo.

una propria e specifica composizione. Di solito, infatti, il test set contiene al suo interno una percentuale di record minore rispetto a quello di training.

Date le dimensioni ridotte del dataset in nostro possesso (1114 frasi) e date le osservazioni fatte in 5.2 circa la necessità di fornire ai sistemi di *Machine Learning* grandi quantità di dati durante la fase di addestramento, per la presente analisi si è deciso di utilizzare una particolare modalità di costruzione dei 2 set in questione: la *k-fold validation*. Questa tecnica è particolarmente utile quando non si possiedono grandissime quantità di dati ma si vuole, comunque, sfruttare il massimo delle loro potenzialità.

La *k-fold validation* prevede, infatti, la suddivisione del dataset in k parti uguali (dette appunto *fold*). Per i volte (dove i è un numero che va da 1 a k), il modello viene addestrato su *k-1* set e testato (o validato) sull'i-esimo fold, ovvero l'unico rimasto escluso dalla fase di training.[17] Ad ogni iterazione, vengono memorizzate le predizioni del modello fatte nell'i-esimo set, al fine di mettere in piedi il *classification report* finale complessivo di tutti i modelli addestrati, di cui si parlerà nel prossimo paragrafo.

La Figura 5.6 sintetizza il processo appena descritto:

1. Il dataset viene scomposto in k-fold;

2. A ogni iterazione i, l'i-esimo set (sfondo bianco) viene messo da parte e utilizzato come test (o validation) set. Tutti gli altri (sfondo blu) vanno a costituire il training set;

3. Per ogni iterazione vengono conservate le predizioni che verranno, a loro volta, usate per calcolare il *classification report* finale.

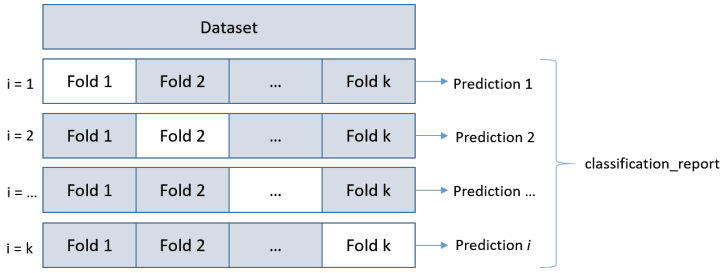


Figura 5.6: Rappresentazione del funzionamento della *k-fold validation*.

Degli specifici criteri tenuti in considerazione per la separazione tra training e test set si parlerà più in dettaglio nei paragrafi 6.3 e 7.3 rispettivamente per il classificatore di *attention* e quello di categorie tematiche.

[17] Come suggerisce il nome di questa tecnica, la *k-fold validation* sarebbe di fatto una tecnica di validazione. Eppure, a seconda delle specifiche esigenze dell'esperimento, si può arbitrariamente decidere di utilizzare l'i-esimo fold come test set piuttosto che come *validation set*.

5.4.1 Normalizzazione dei dati

Prima di fornire un insieme di dati a un modello di classificazione automatico, di solito questi vengono opportunamente processati. Una trasformazione molto comune prevede la standardizzazione, o meglio *normalizzazione*, dei dati in ingresso.[18]

Per *standardizzazione* si intende la trasformazione dei valori di tutte o parte delle variabili di un dataset in una scala di valori comune. Questa trasformazione è necessaria al fine di risolvere il problema relativo all'impossibilità di paragonare attributi aventi scale differenti.

Un esempio banale, ma al contempo esplicativo, è il seguente: si immagini di avere un gruppo di persone, ognuna descritta da feature quali altezza e peso (rispettivamente misurate in metri e kilogrammi). Sarebbe impossibile paragonare i record del dataset in questione senza che una delle due variabili prevalga sull'altra, in quanto una delle due avrà inevitabilmente valori (e di conseguenza margini di variazione) maggiori rispetto all'altra.

Nel nostro caso, l'operazione di normalizzazione è stata effettuata utilizzando la classe *MinMaxScaler* del modulo *preprocessing* della libreria *sklearn*.[19] Le classe serve per istanziare un oggetto sul quale è possibile invocare la funzione *fit_transform* che permette, con un'unica operazione, di fittare i dati e trasformarli restituendo l'output desiderato.

$$scaler = MinMaxScaler()$$
$$processed\_df.loc[:, :] = scaler.fit\_transform(processed\_df.values)$$

In particolare, l'operazione di normalizzazione adottata ha portato tutte le features in un range di valori compreso tra 0 e 1.

La Figura 5.7a mostra le prime 10 righe del dataset non normalizzate in corrispondenza delle feature *tokens_per_sent*, *char_per_tok* e *upon_dist_ADJ*. La Figura 5.7b mostra, invece, l'aspetto delle stesse righe e colonne a seguito dell'applicazione del *MinMaxScaler*.

tokens_per_sent	char_per_tok	upos_dist_ADJ
6	5,75	0
25	4,69565217391304	0
5	4,25	0
8	2,4	0
4	6,66666666666667	25
2	2	0
2	2	0
21	4,85	9,52380952380952
21	4	4,76190476190476
24	4,1304347826087	0

(a) Colonne non normalizzate.

tokens_per_sent	char_per_tok	upos_dist_ADJ
0,049382716049383	0,416666666666667	0
0,283950617283951	0,29951690821256	0
0,037037037037037	0,25	0
0,074074074074074	0,044444444444444	0
0,024691358024691	0,518518518518518	0,5
0	0	0
0	0	0
0,234567901234568	0,316666666666667	0,19047619047619
0,234567901234568	0,222222222222222	0,095238095238095
0,271604938271605	0,23671497584541	0

(b) Colonne normalizzate.

Figura 5.7: Esempio di normalizzazione.

[18] Questa precisione terminologica è d'obbligo per evitare di confondere la normalizzazione (intesa come *standardizzazione*) con la normalizzazione, tanto usata in statistica, intesa come trasformazione di una variabile al fine di ottenere la sua distribuzione *gaussiana* (o *normale*).

[19] Pagina di riferimento: `https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html`.

5.5 Metriche di valutazione

Si è detto che un altro dei momenti fondamentali dopo la costruzione di un classificatore è il momento della sua valutazione (*performance evaluation*). Anche in questo la libreria di Python *sklearn* si mostra di estrema utilità in quanto fornisce una funzione in grado di raggruppare diverse metriche volte a valutare la qualità dell'annotazione prodotta. Si vuole qui di seguito, infatti, offrire una panoramica di quali siano le informazioni fornite dalla funzione *classification_report* del modulo *metrics* di *sklearn*.

Un esempio dell'output fornito dalla funzione è visibile in Tabella 5.1.

	precision	recall	f1-score	support
engagement	0.60	0.50	0.54	565
no_engagement	0.55	0.64	0.59	535
accuracy			0.57	1100
macro avg	0.57	0.57	0.57	1100
weighted avg	0.57	0.57	0.57	1100

Tabella 5.1: Un esempio di *classification report* nel caso di una classificazione binaria che preveda l'assegnazione delle categorie *engagement* e *no_ engagement*.

Per meglio comprendere il contenuto del report è necessario, prima, introdurre le definizioni di *Veri Positivi*, *Veri Negativi*, *Falsi Positivi* e *Falsi Negativi*:

- TP (*True Positive*) e TN (*True Negative*): dati che sono stati etichettati correttamente dal modello (rispettivamente valori positivi etichettati come positivi e valori negativi etichettati come negativi);
- FP (*False Positive*) e FN (*False Negative*): dati che sono stati etichettati con una label errata (rispettivamente valori negativi etichettati come positivi e valori positivi etichettati come negativi).

	Predicted Class = Yes	Predicted Class = No
True Class = Yes	TP	FN
True Class = No	FP	TN

Tabella 5.2: *Confusion matrix* per problemi di classificazione binari.

La matrice di confusione[20] visibile in Tabella 5.2 è una tabella le cui righe rappresentano le *true label* degli elementi e le cui colonne corrispondono alle categorie predette. In altre parole, in una *confusion matrix*, è possibile identificare il numero di TP, FN, FP e TN e vedere la loro distribuzione.

[20]Il particolare nome di *matrice di confusione* deriva dal fatto che la tabella fa vedere in quale misura l'algoritmo "si confonde" attribuendo a un dato elemento una categoria diversa rispetto a quella di appartenenza.

Ogni entrata specifica, infatti, il numero di elementi di una data classe che sono stati classificati con la classe indicata nella rispettiva colonna. Idealmente, un classificatore perfetto (con nessun errore di classificazione), avrebbe delle entrate diverse da 0 solo ed esclusivamente lungo la diagonale.

Sklearn fornisce una funzione predefinita del modulo *metrics* (*confusion_ matrix)* che, prendendo in input, l'insieme di etichette reali e predette, è in grado di fornire come output la matrice di confusione stessa.

$$cm = confusion\_matrix(true\_labels,\ predicted\_labels)$$

A partire dai valori contenuti nella matrice di confusione, la funzione *classification_ report* calcola alcune metriche tradizionalmente usate per valutare le performance di un algoritmo di classificazione. In particolare:

- *Accuracy*: sta a indicare la frazione di tutte le predizioni che il modello ha correttamente identificato;

$$Accuracy = \frac{TP + TN}{(TP + TN + FP + FN)}$$

- *Precision*: indica il grado di affidabilità delle predizioni positive;

$$Precision = \frac{TP}{(TP + FP)}$$

- *Recall*: indica la frazione di positivi che sono stati correttamente identificati dal modello;

$$Recall = \frac{TP}{(TP + FN)}$$

- *F1-score* o *armonic mean*: si calcola come la media pesata di precision e recall.

$$F1 - score = \frac{2 * (Precision * Recall)}{(Precision + Recall)}$$

Nel report, tuttavia, è possibile trovare anche alcune metriche pesate a cui, di solito, si fa riferimento nei casi in cui il *training data* si mostri sbilanciato verso una delle due categorie possibili:

- *Micro avg*;
- *Macro avg*;
- *Weighted avg.*

Un eventuale sbilanciamento del corpus di dati si può notare dai valori della colonna *support* che mostra il numero di elementi (forniti in fase di testing) facenti capo all'una o all'altra categoria. Nel caso dell'esempio riportato in Tabella 5.1, il *test set* si presenta, in realtà, abbastanza bilanciato nella distribuzione dei valori tra le classi (565 record per la classe *engagement* e 535 per quella *no_ engagement*).

5.5.1 Il concetto di *baseline*

Avere delle metriche, quali quelle sopra descritte, utili a misurare la performance di un modello è essenziale per poter procedere a una sua valutazione. Eppure per poter capire se, di fatto, le performance di un modello siano accettabili o meno, è necessario confrontare i risultati ottenuti con quella che, in gergo tecnico, viene definita una *baseline*.

In generale, la *baseline* si presenta come un *benchmark*, ovvero un modello di riferimento al quale guardare per comprendere se il nostro sistema stia introducendo un qualche miglioramento rispetto allo stato dell'arte e, se sì, in quale proporzione. Un modello si potrà ritenere interessante (e meritevole di essere indagato più a fondo) nel caso in cui i valori di performance mostrati superino, appunto, quelli della *baseline*.

Nel caso di task mai eseguiti prima (come quelli di cui si parlerà nei prossimi capitoli) per i quali non esiste uno stato dell'arte, non ci si confronta con quest'ultimo bensì con una *baseline*, per così dire, randomica calcolata come il rapporto tra il numero di occorrenze del fenomeno più frequente e il numero di elementi totale. Da qui si comprende, dunque, come la *baseline* possa identificare, oltre che un modello rappresentante lo stato dell'arte, un generico modello che, in presenza di nuovi dati, assegnerebbe solo ed esclusivamente la label che occorre con maggiore frequenza.

5.6 Lo studio del modello e la selezione delle feature

L'ultimo step del presente studio ha previsto l'individuazione di un set significativo di feature. La fase cosiddetta di *feature selection* prevede la definizione del sottoinsieme di caratteristiche dotate di un maggiore potere predittivo.

Nella comunità scientifica è risaputo che l'individuazione di uno spazio dimensionale ridotto ha una serie di vantaggi, tra cui il miglioramento delle performance del predittore (nel caso di una nuova fase di addestramento con il sottoinsieme individuato), la riduzione dei costi temporali per la fase di training e una maggiore comprensione del fenomeno che i dati rappresentano.[21]

Per far questo, si è proceduto ad addestrare ogni modello di classificazione sulla totalità degli elementi in possesso, senza, cioè, passare per la fase di testing.[22] Dopo aver addestrato i modelli secondo questa modalità, si sono sfruttate due funzioni della libreria *sklearn*: nel caso di SVM, la funzione *coef_* e, nel caso di Random Forest, *feature_importances_*. Entrambe le funzioni menzionate restituiscono un array contenente i pesi associati a ogni feature. I pesi sono quelli utilizzati dal classificatore e indicano quanto decisiva sia una feature nel discriminare un elemento di una classe da quelli di altre classi.

[21] Per un maggiore approfondimento si rimanda a Guyon e Elisseeff (2003).

[22] Questa modalità di procedere non è affatto scontata dal momento che, come anche detto *supra* (par. 5.4), il dataset originario viene solitamente scomposto a formare due insiemi di dati distinti, su cui effettuare, rispettivamente, una fase di training e una di test.

In questa maniera è stato possibile indagare più a fondo il funzionamento dei classificatori e provare a motivare le scelte fatte da ognuno di essi. Un'analisi più accurata verrà condotta nei prossimi capitoli facendo strettamente riferimento al comportamento di due classificatori individuati rispettivamente per ognuno dei fenomeni di interesse.

Capitolo 6

Studio del fenomeno di attenzione

L'obiettivo di questo capitolo è quello di fornire una possibile applicazione pratica del corpus di frasi di lingua parlata costruito seguendo la pipeline descritta nel capitolo 4. Prima di darlo in pasto ai vari modelli di classificazione, il corpus verrà ulteriormente processato al fine di renderlo adeguato per l'addestramento di sistemi di classificazione automatici.

In particolare, verranno descritti ed eseguiti diversi esperimenti basati su 3 scenari di classificazione. Per ognuno di questi verranno utilizzati 2 modelli di classificazione (SVM e Random Forest) ai quali si forniranno in input 6 dataset contenenti, al loro interno, spazi dimensionali differenti.

Si commenteranno, infine, i risultati ottenuti per cercare di comprendere il meccanismo di funzionamento dei vari sistemi e cercare di rispondere a domande di interesse generale come:

- Sulla base del modo in cui una frase è linguisticamente composta o prosodicamente pronunciata, è possibile prevedere se questa genererà *attention*?[1]
- Se esistono feature linguistiche e prosodico-acustiche discriminanti il fenomeno di attenzione, quali sono queste feature e come si influenzano reciprocamente?

6.1 Le caratteristiche dei dati

Come emerso dal capitolo 3, la costruzione di un classificatore di *attention* non costituiva una novità assoluta tra i numerosi esperimenti condotti presso l'Istituto di Linguistica Computazionale di Pisa. Eppure, la presente analisi si discosta da quelle precedenti sotto diversi punti di vista.

Rispetto allo studio di Poggianti, di cui si è parlato nel capitolo 3, vi è:

[1] Per capire cosa si intende per *attention* si rimanda *supra* alla sezione 3.1.

- Un maggiore spazio dimensionale. Al corpus sono state, infatti, aggiunte più feature (tra cui quelle acustiche);
- Una maggiore affidabilità in termini di segmentazione del testo, in quanto il processo di *speech segmentation* è stato effettuato manualmente e confrontando l'annotazione con un secondo annotatore.

Rispetto allo studio di Boggia, delle cui caratteristiche si è, invece, accennato nella sezione 4.1, vi è:

- Un maggiore numero di dati da analizzare, in quanto sono state da me acquisite 3 visite invece di 2;
- L'annotazione dell'informazione tematica per ogni frase del dataset.

Come diretta conseguenza dell'aggiunta di quest'ultima informazione, i dati CHROME sono stati utilizzati per la costruzione di un secondo classificatore che si descriverà nel capitolo 7.

6.2 Trasformazioni preliminari del dataset

Prima ancora di eseguire gli esperimenti di cui si parlerà nel paragrafo seguente, è stato necessario effettuare una serie di manipolazioni al dataset in aggiunta a quelle già descritte nel capitolo 4.

Il dataset contenente tutte le feature necessitava, innanzitutto, di essere normalizzato. Circa l'importanza e le modalità di esecuzione di questo step si è già detto nel paragrafo 5.4.1.

Un'ulteriore trasformazione è stata la scomposizione della colonna delle categorie tematiche in 8 colonne distinte, ciascuna corrispondente a uno degli 8 valori possibili. A seconda della categoria di appartenenza, la frase avrebbe avuto, in corrispondenza della rispettiva colonna, il valore 1 e il valore 0 in tutte le altre.

Un esempio di ciò è mostrato in Figura 6.1: la colonna *category* presenta i valori delle categorie così come sono stati annotati durante la fase descritta in 4.2.6; nelle colonne sulla destra, invece, è possibile vedere la diversa modalità di annotazione utilizzata in questa fase. Ad esempio, la frase della prima riga appartiene alla categoria F. Ciò significa che, nella nuova annotazione, si avrà il valore 1 sotto la colonna *catF* e il valore 0 in tutte le altre.

Il vantaggio di avere le informazioni tematiche così annotate risiedeva nella possibilità di poter individuare, in fase di studio del modello, esattamente quale, tra le tante categorie, contribuisse in maniera più o meno decisiva alla sua definizione. Per l'aggiunta delle colonne in questione si è realizzato un apposito script in Python *script_ classificatore1_ creaColonneCategorie.py* consultabile in allegato.

category	catA	catB	catC	catD	catE	catF	catG	catH
F	0	0	0	0	0	1	0	0
F	0	0	0	0	0	1	0	0
F	0	0	0	0	0	1	0	0
F	0	0	0	0	0	1	0	0
F	0	0	0	0	0	1	0	0
F	0	0	0	0	0	1	0	0
F	0	0	0	0	0	1	0	0
H	0	0	0	0	0	0	0	1
H	0	0	0	0	0	0	0	1
G	0	0	0	0	0	0	1	0
A	1	0	0	0	0	0	0	0

Figura 6.1: Trasformazione dell'informazione tematica.

6.3 Gli scenari di classificazione

Una delle sfide, quando si costruisce un sistema di classificazione, è trovare un modello che abbia delle performance soddisfacenti che superino, con un margine abbastanza ampio, quelle della *baseline*.[2] Uno dei modi migliori per esplorare il campo delle opzioni possibili, è quella di condurre diversi esperimenti.

Nel caso di questo particolare studio la scelta di selezionare 2 modelli di apprendimento, SVM e Random Forest, è giustificata dalla conseguente possibilità di disporre di 2 set di feature più significative. Individuare eventuali feature comuni a entrambi i classificatori avrebbe rafforzato l'ipotesi per cui esistono delle feature effettivamente determinanti il fenomeno di attenzione.

Oltre a provare ad analizzare il comportamento di 2 diversi modelli di classificazione l'obiettivo, però, era anche quello di comprendere le possibili relazioni e dipendenze tra feature differenti. Per tale ragione si è deciso di addestrare i modelli selezionati con dataset aventi spazi dimensionali differenti.

Si è proceduto, pertanto, a ricavare dal dataset, per così dire, "originario" contenente tutte le feature, altri 5 dataset, per un totale di 6. Ognuno di questi presenta i nomi e le caratteristiche mostrate in Tabella 6.1.

Nome dataset	**Caratteristiche**
all-feats	Contenente tutte le feature (linguistiche, acustiche e relative alle categorie tematiche)
ling-feats	Contenente esclusivamente le feature linguistiche
ling-feats-withCat	Contenente le feature linguistiche con l'aggiunta delle categorie tematiche
acoust-feats	Contenente esclusivamente le feature acustiche
acoust-feats-withCat	Contenente le feature acustiche con l'aggiunta delle categorie tematiche
categories	Contenente esclusivamente le informazioni tematiche

Tabella 6.1: Nomi e caratteristiche dei 6 dataset costruiti.

[2] Cfr. *supra*, par. 5.5.1.

La costruzione di questi dataset è avvenuta in maniera automatica mediante lo script *script_ classificatore1_ estraiDataset.py*, allegato alla Tesi.

I classificatori sono, infine, stati addestrati secondo 3 diverse modalità di *k-fold validation*,[3] ognuna delle quali presenta un diverso valore di k nonché un differente criterio di selezione degli elementi del dataset che sarebbero andati a comporre rispettivamente training e test set. Di questa tipologia di esperimenti si parlerà più in dettaglio nelle prossime sottosezioni dedicate.

6.3.1 Classificazione randomica: *11-fold validation*

Il primo esperimento di classificazione ha previsto quella che si potrebbe definire una *11-fold cross validation*. Sebbene, infatti, la pratica più diffusa sia quella di suddividere il dataset in 10 parti (ovvero di effettuare una *10-fold cross validation*), essendo il nostro dataset composto da 1114 frasi, la cosa più logica da fare era quella di suddividerlo in 11 parti composte da 100 frasi ciascuna facendo in modo che, a ogni iterazione, le 14 frasi rimanenti fossero inglobate all'interno del training set.

Un'idea più chiara di quanto detto è fornita dalla Figura 6.2 che mostra, nella seconda riga, la composizione di ciascuno degli 11 fold in cui il dataset è stato scomposto, più la composizione del 12esimo fold formato da sole 14 frasi. A ogni iterazione, queste ultime sono sempre inglobate nel training set, rappresentato dalla somma di tutti i fold con lo sfondo blu.

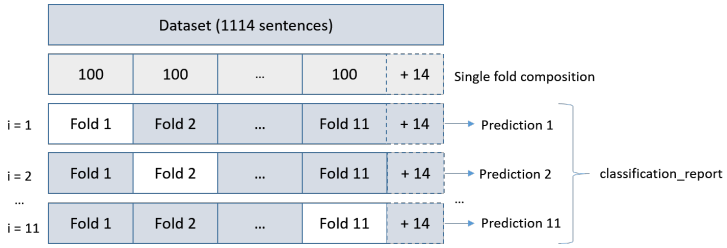


Figura 6.2: *11-fold cross validation*.

Come risultato di questa operazione, il numero di predizioni finali effettuate dal modello, e su cui è stato costruito il *classification report*, sono solamente 1100.

Prima di suddividere il dataset secondo le modalità sopra descritte, è stato, però, necessario effettuare due operazioni.

In primis, essendo le frasi all'interno del dataset riordinate per visita e per *Point of Interest*, si è dovuto procedere con un'operazione di randomizzazione al fine di garantire una maggiore variabilità interna ai set. Per far questo si è utilizzata la funzione *sample*, richiamabile su un *DataFrame pandas*,[4] come mostrato qui di seguito:

[3] Cfr. *supra*, par. 5.4.

[4] *Pandas* è una libreria *open-source* di Python che fornisce una serie di strutture dati e strumenti di analisi semplici e facili da utilizzare. Tra le struttura dati più utilizzate vi sono,

shuffled_df = df.sample(len(df), replace=False)

L'altra operazione, successiva alla randomizzazione, è stata la rimozione di colonne contenenti informazioni categoriche o non rilevanti ai fini dell'analisi. In particolare le colonne: *file*, *id*, *start*, *end* e *text*. Si è deciso di eliminare le colonne in questa fase dell'analisi poiché questo avrebbe dato la possibilità di ripristinarle nel dataset di output, contenente le predizioni del modello, rendendolo più leggibile agli occhi di un utente esterno.

Gli 11 set da 100 frasi, insieme a quello composto da 14, sono stati, infine, ottenuti mediante una semplice operazione di slicing del dataset processato. Questi set sono stati alla base delle fasi di training e di test dei modelli prescelti (ovvero SVM e Random Forest) seguendo le modalità previste dalla tradizionale tecnica della *k-fold validation* descritta nel paragrafo 5.4.

6.3.2 Classificazione per POI: *POI-fold validation*

Il secondo scenario di classificazione possibile (al pari di quello di cui si parlerà nel prossimo sottoparagrafo) era attuabile grazie alla struttura intrinseca dei dati in possesso.[5] Vi era la possibilità, infatti, di poter raggruppare le frasi a seconda del *Point of interest* di appartenenza. Ciò significava che il classificatore sarebbe stato addestrato "orizzontalmente" su tutti i vari luoghi di interesse della visita tranne uno.

La Figura 6.3 mostra cosa si intende per "addestramento orizzontale": ogni fold prevede il raggruppamento di parti, per così dire, corrispondenti ma di visite diverse. Nell'immagine è rappresentato un ipotetico scenario di iterazione 2 ($i = 2$) in cui il fold 2 (composto da tutti i quadratini con sfondo bianco) è utilizzato come test set, mentre tutti gli altri (sfondo blu) come training set.

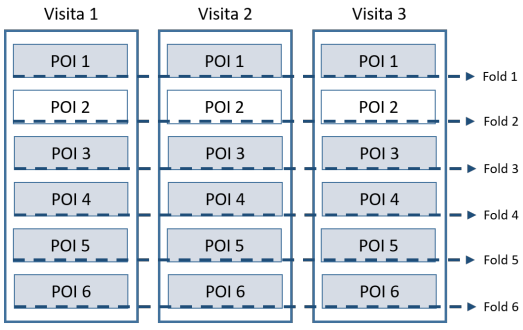


Figura 6.3: *POI-fold validation*: addestramento orizzontale.

per l'appunto, i dataframe. Link alla documentazione Pandas: `https://pandas.pydata.org/pandas-docs/stable/index.html`. Link alla documentazione DataFrame: `https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.html`.

[5] Cfr. *supra*, Tabella 4.7.

In questo tipo di addestramento si presumeva che la difficoltà consistesse nella necessità di dover prevedere etichette di frasi appartenenti a un contesto totalmente nuovo.

Se nel primo caso era stato possibile dividere il dataset in 11 fold su cui iterare per le fasi di training e validazione, in questo caso i fold possibili si riducevano a un numero di 6.

Come nel caso dell'*11-fold cross validation*, però, anche qui si rendeva necessario eliminare le colonne contenenti informazioni categoriche o non rilevanti ai fini dell'analisi (*file*, *id*, *start*, *end* e *text*) prima di procedere alla suddivisione del dataset nei 6 fold in questione.

6.3.3 Classificazione per visita: *visit-fold validation*

Il terzo e ultimo scenario di classificazione è la suddivisione e il raggruppamento delle frasi del dataset a seconda della visita di appartenenza. Sulla base di quanto detto nel paragrafo precedente, si potrebbe, in tal caso, parlare, di addestramento "verticale" : il classificatore è stato, infatti, addestrato su 2 visite e testato su 1 differente.

La Figura 6.4 dà un'idea più chiara di cosa si intenda per "addestramento verticale", ovvero il raggruppamento di segmenti (POI) diversi appartenenti, però, alla stessa visita (in termini di *k-fold validation*, la *visit-fold validation* avrebbe, perciò, significato la presenza di 3 set, ciascuno contenente frasi appartenenti a una stessa visita). Nell'immagine è rappresentato un ipotetico scenario di iterazione 2 ($i = 2$) in cui il fold 2 (composto da tutti i quadratini con sfondo bianco) è utilizzato come test set, mentre tutti gli altri (sfondo blu) come training set.

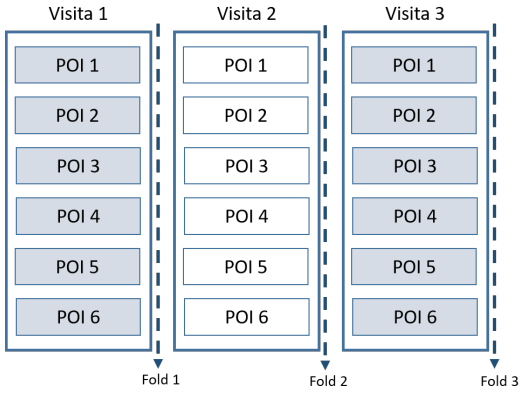


Figura 6.4: *Visit-fold validation*: addestramento verticale.

Come nei due scenari precedenti, anche qui la rimozione temporanea delle colonne *file*, *id*, *start*, *end* e *text* era necessaria per procedere all'addestramento del modello.

6.4 Risultati degli esperimenti

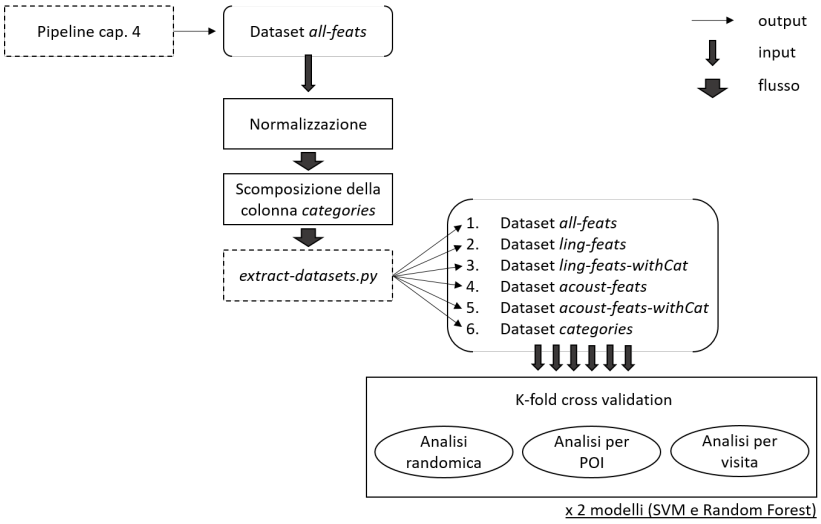


Figura 6.5: Visualizzazione del flusso descritto nei paragrafi 6.2 e 6.3.

Una volta fatti girare tutti i modelli, con le diverse combinazioni di dati e feature, si sono raccolti i *classification report* di tutti gli algoritmi addestrati al fine di confrontare i risultati prodotti.[6]

Il numero totale degli esperimenti condotti è pari a 36, come si può vedere dallo schema riportato in Figura 6.5: 6 dataset vengono utilizzati per 3 diversi esperimenti, ognuno dei quali viene fatto girare 2 volte con i modelli di classificazione prescelti (SVM e Random Forest).

Per poter valutare le performance era, innanzitutto, necessario avere un'idea chiara di quale fosse la *baseline*, ovvero il valore sopra il quale un modello può essere ritenuto accettabile. Come anche detto in 5.5.1, la *baseline* si calcola dividendo il numero di occorrenze del fenomeno più frequente (solitamente il numero di frasi annotate con un valore di *attention* positivo) con il numero di elementi totale (che varia a seconda dell'esperimento condotto).

Nel caso in cui si consideri la totalità delle frasi del dataset, la *baseline* è calcolata come segue:

$$Baseline = \frac{565}{1114} * 100 = 50,7\% \approx 51\%$$

Il valore sopra riportato può ritenersi una buona approssimazione dei diversi valori di *baseline* corrispondenti ai vari esperimenti, in quanto essi oscillano solitamente tra il 51% e il 52%.[7]

[6] Tutti i report sono visionabili in Appendice A.1.

[7] Essendo i valori di *baseline* diversi per ogni esperimento, si rimanda all'Appendice A.1 per la verifica.

Essendo molti gli esperimenti condotti, in questa fase di analisi dei risultati ci si limiterà a commentare esclusivamente i valori di accuratezza che comunque si presenta come una metrica abbastanza affidabile per la valutazione dei sistemi in questione in quanto i dati si distribuiscono in maniera quasi del tutto bilanciata entro le 2 categorie *engagement* e *no_ engagement*.[8]

Sulla base della *overview* offerta dalla Tabella 6.2, si osserva che molti valori di accuratezza sono superiori alla *baseline*, seppure alcuni non se ne discostino abbastanza.

SVM	***11-fold***	***POI-fold***	***visit-fold***
all-feats	61 %	58 %	55 %
ling-feats	57 %	56 %	58 %
ling-feats-withCat	58 %	57 %	56 %
acoust-feats	57 %	59 %	57 %
acoust-feats-witCat	57 %	59 %	56 %
categories	50 %	52 %	49 %

RANDOM FOREST	***11-fold***	***POI-fold***	***visit-fold***
all-feats	60 %	62 %	60 %
ling-feats	59 %	60 %	58 %
ling-feats-withCat	60 %	59 %	57 %
acoust-feats	59 %	59 %	59 %
acoust-feats-witCat	60 %	61 %	59 %
categories	51 %	52 %	49 %

Tabella 6.2: Valori percentuali di accuratezza rispettivamente per SVM e Random Forest.

In generale, si nota che gli addestramenti fatti con il dataset *ling-feats* (solo feature linguistiche) e *acoust-feats* (solo feature acustiche) non presentano, tra loro, grandi variazioni in termini di performance. Mediamente l'accuratezza dei modelli addestrati su ognuno dei dataset sopra menzionati è rispettivamente del 58% (per il dataset *ling-feats*) e 58,3% (per il dataset *acoust-feats*).

L'addestramento fatto, invece, sui dataset *ling-feats-withCat* (feature linguistiche + categorie tematiche) e *acoust-feats-withCat* (feature acustiche + categorie tematiche) fa rimanere quasi del tutto invariata la performance dei modelli addestrati: rispetto ai casi precedenti in alcuni scenari si verifica un miglioramento, in altri un peggioramento delle prestazioni del sistema, stando ciò a significare che l'aggiunta dell'informazione, per così dire, tematica non contribuisce, di fatto, a migliorare in maniera sostanziale la conoscenza che il sistema ha riguardo il fenomeno di interesse.

[8] La distribuzione varia a seconda di quale sia l'esperimento. Per i risultati precisi si rimanda, dunque, all'Appendice A.1 e, in particolare, alla colonna *support* dei vari *classification report*. In generale, però, nel dataset completo di 1114 frasi, vi sono 565 frasi annotate con l'etichetta *engagement* e 549 con *no_ engagement*.

Una simile osservazione sembra essere avvalorata sia dal posizionamento che le informazioni categoriali presentano nel ranking delle feature prodotto a seguito di questi addestramenti[9] sia dai bassi valori di accuratezza che si registrano per il dataset *categories* in pressoché tutti gli esperimenti (in cui si registrano anche dei casi al di sotto della *baseline*).

L'addestramento fatto, invece, sul dataset *all-feats* (feature linguistiche e acustiche + categorie) presenta, seppur di poco, valori migliori registrati mediamente in quasi tutti gli addestramenti. Ciò ha come conseguenza la supposizione per cui un dataset multimodale, contenente al suo interno sia informazioni linguistiche che prosodico-acustiche, sia quello che, in un certo qual modo, debba essere analizzato in maniera più approfondita per cercare di raggiungere gli scopi prefissati in questo studio.

Per tale ragione si è condotto su di esso lo studio di cui al paragrafo 6.5.1. In specifico, verrà analizzato il ranking delle feature prodotto dal modello SVM addestrato con una *11-fold validation* sul dataset *all-feats*.

6.5 Riaddestramento del classificatore su un subdataset multimodale

Una volta scelto il modello su cui effettuare lo studio, oltre che procedere a un'osservazione più approfondita del suo funzionamento (andando ad esaminare le tipologie di feature maggiormente tenute in considerazione durante la fase di predizione), si è deciso di procedere a una nuova fase di addestramento del modello fornendo in input un set di feature *ad hoc*.

La ragione di questo riaddestramento risiede, *in primis*, nel fatto che le performance ottenute, in generale, in tutti gli esperimenti[10] non mostrano un significativo distacco rispetto ai valori di *baseline*. Risultati di performance non ottimali potrebbero essere causati da un eccessivo sbilanciamento tra il numero di feature fornite (323 nel caso del dataset *all-feats*) rispetto al numero di record costituenti il dataset nella sua totalità (1114). Come anche accennato nella sezione 5.6, il riaddestramento di un modello con uno spazio dimensionale ridotto avrebbe potuto portare a un miglioramento delle performance del predittore.

Inoltre, poiché il set di feature da fornire al sistema in questa nuova fase sarebbe stato selezionato a partire dal ranking prodotto a seguito degli addestramenti fatti sui dataset *ling-feats* (solo feature linguistiche) e *acoust-feats* (solo feature acustiche), il nuovo riaddestramento avrebbe aiutato a comprendere maggiormente le interrelazioni presenti tra le due diverse tipologie di feature: linguistiche da un lato e acustiche dall'altro.

6.5.1 Analisi e selezione delle feature

Al fine di selezionare un insieme di feature significative da fornire come input nella nuova fase di addestramento, si è proceduto ad analizzare l'output della

[9] Tutti i ranking delle feature suddivisi per addestramento sono stati allegati alla Tesi.
[10] Cfr. *supra*, Tabella 6.2.

fase di studio del modello descritta nel paragrafo 5.6 che, si ricordi, prevedeva l'invocazione di una funzione specifica a seconda del tipo di classificatore: la funzione *coef_* per SVM e la funzione *feature_importances_* per Random Forest. Ognuna di queste prevedeva l'assegnazione, a ogni feature, del relativo peso (indice della sua importanza per uno specifico modello). L'output di queste funzioni è stato opportunamente salvato in un file csv.

In particolare, sono stati analizzati i pesi associati da SVM in 2 casi: rispettivamente nel caso di addestramento con i dataset *ling-feats* e *acoust-feats*.[11] L'esclusione delle informazioni tematiche è giustificata dal fatto che il dataset *categories* ha mostrato le performance peggiori tra tutti i possibili dataset, come osservato nella Tabella 6.2.

Le 20 migliori feature linguistiche e acustiche, con i rispettivi pesi, sono mostrate nei grafici delle Figure 6.6 e 6.7.

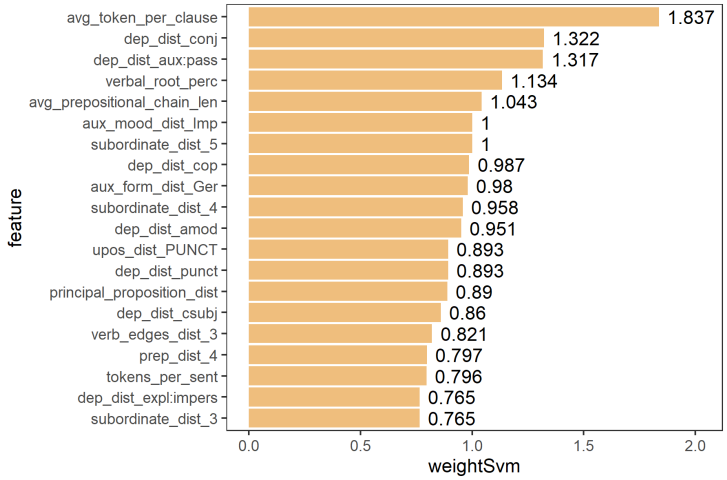


Figura 6.6: La 20 feature linguistiche più significative per SVM.

Tra le top 20 feature linguistiche, la maggior parte sono informazioni sintattiche (16), seguite da informazioni morfosintattiche (3) e 1 sola feature relativa alle proprietà del testo grezzo (Cfr. Tabella 6.3).

La maggior parte delle feature sintattiche sono del tipo *dep_dist_** relative alla distribuzione delle 37 relazioni usate in *Universal Dependencies*. Anche la feature linguistica in assoluto più significativa è di tipo sintattico: l'*avg_token_per_clause* rappresenta, infatti, la lunghezza media per frase calcolata in termini di numero di token.[12]

[11] I ranking completi di questi due casi possono essere consultati rispettivamente nelle Appendici A.2.1 e A.2.2.

[12] Nella *profiling-legend*, allegata alla Tesi, l'*avg_token_per_clause* è definita come segue: «average clause length, calculated in terms of the average number of tokens per clause, where a clause is defined as the ratio between the number of tokens in a sentence and the number of either verbal or copular head» («lunghezza media di frase calcolata in termini di numero

	Group
SYNTACTIC FEATURES	
avg_token_per_clause	Global and Local Parsed Tree Structures
dep_dist_conj	Syntactic Relations
dep_dist_aux:pass	Syntactic Relations
verbal_root_perc	Verbal Predicate Structure
avg_prepositional_chain_len	Global and Local Parsed Tree Structures
subordinate_dist_5	Use of Subordination
dep_dist_cop	Syntactic Relations
subordinate_dist_4	Use of Subordination
dep_dist_amod	Syntactic Relations
dep_dist_punct	Syntactic Relations
principal_proposition_dist	Use of Subordination
dep_dist_csubj	Syntactic Relations
verb_edges_dist_3	Verbal Predicate Structure
prep_dist_4	Global and Local Parsed Tree Structures
dep_dist_expl:impers	Syntactic Relations
subordinate_dist_3	Use of Subordination
MORPHOSYNTACTIC INFO	
aux_mood_dist_Imp	Inflectional morphology
aux_form_dist_Ger	Inflectional morphology
upos_dist_PUNCT	-
RAW TEXT PROPERTIES	
tokens_per_sent	-

Tabella 6.3: Categoria di appartenenza delle 20 migliori feature secondo SVM.

Tra le feature acustiche, invece, vi sono prevalentemente informazioni afferenti alla categoria *Spectral LLD* che identifica informazioni riguardanti lo spettro acustico. In particolare, 8 sono MFCC (*Mel Frequency Cepstrum Coefficients*), ovvero coefficienti dell'MFC.[13]

Uno dei coefficienti in questione (*mfcc_sma[3]_std*) si colloca in prima posizione approssimativamente con un peso di 1.99, aggiudicandosi la feature più rilevante in assoluto tra quelle acustiche.

Oltre ad informazioni relative allo spettro, tra le migliori feature acustiche ve ne sono anche alcune afferenti alle altre categorie identificate in 4.2.5: in

medio di token per frase, dove una frase è definita come il rapporto tra il numero di token in una frase e il numero di teste verbali o copule»).

[13]La *Mel-frequency cepstrum* è una particolare tipologia di rappresentazione dello spettro dell'energia di un suono molto usata nel campo dello *speech recognition*. Per maggiori approfondimenti si può consultare Logan (2000).

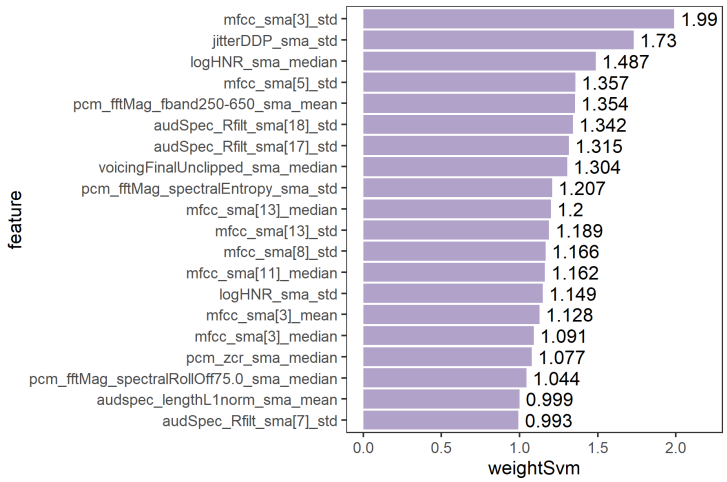


Figura 6.7: La 20 feature acustiche più significative per SVM.

specifico, 4 informazioni relative al suono della voce (*Voicing related LLD*) e 2 relative all'energia (*Energy related LLD*).

Si può notare che le feature acustiche presentano valori più elevati rispetto a quelle linguistiche, stando a significare che esse sono ritenute più importanti da SVM.[14]

6.5.2 La fase di riaddestramento

Identificate, quindi, le 40 feature linguistiche e acustiche più significative si è messo a punto lo script *script_ classificatore1_ estrai_ datasetTop40* in grado di prendere in input il dataset *all-feats* e di fornire in output un dataset dal nome *dataset-top40* con uno spazio dimensionale ridotto. Quest'ultimo è stato dato in pasto a SVM secondo la stessa modalità di addestramento descritta in 6.3.1.

A seguito di questa operazione, la performance di SVM non presenta alcuna variazione rispetto all'addestramento precedente effettuato con il dataset *all-feats*, smentendo, dunque, nel nostro caso, l'ipotesi per cui un riaddestramento su un dataset, ripulito da rumore (ovvero feature non rilevanti), conduca a un miglioramento delle prestazioni del sistema.

Seppure il valore di accuratezza rimanga costante (61%), si può comunque notare una differenza nel ranking delle feature prodotto a seguito dell'addestramento. Dalla Figura 6.8 si osserva, infatti, che diverse feature linguistiche (come ad esempio *prep_ dist_ 4*, *tokens_ per_ sent*, *dep_ dist_ csubj*, *principal_ proposition_ dist*) che si posizionavano tra le ultime posizioni nel grafico di Figura 6.6, si ritrovano, adesso, tra le prime posizioni.

[14] Anche questo dato si ritrova in Boggia (2021).

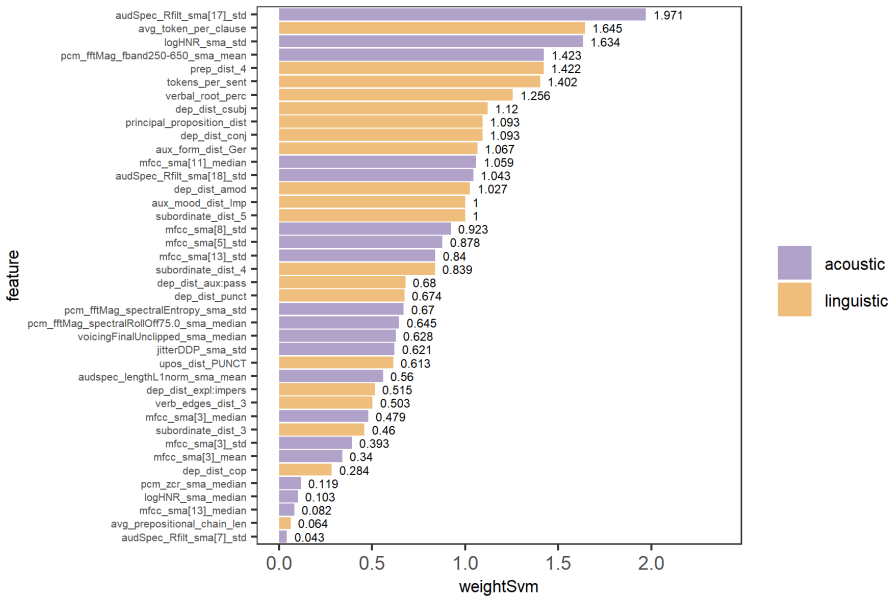


Figura 6.8: Ranking 40 feature linguistiche e acustiche prodotto da SVM a seguito di un riaddestramento su un sub-dataset multimodale.

La feature linguistica *avg_token_per_clause* si riconferma, inoltre, una feature molto rilevante nella discriminazione del fenomeno di attenzione collocandosi, in questo nuovo riaddestramento, in seconda posizione con un punteggio di circa 1.645.

Se nel caso dei due addestramenti separati, rispettivamente per le feature linguistiche e acustiche, queste ultime hanno rivelato punteggi di gran lunga superiori rispetto alle prime, nel caso di un riaddestramento sul sub-dataset multimodale, comprendente sia le feature linguistiche che acustiche, tra le prime top-10 si collocano solo 2 acustiche. Nonostante ciò, la feature in assoluto più rilevante è proprio una informazione relativa allo spettro acustico (*audSpec_Rfilt_sma[17]_std*).

Da notare, invece, che l'informazione spettrale (*audSpec_Rfilt_sma[7]_std*) processata mediante un filtro specifico dal nome RASTA (*Relative Spectral Transform*)[15] si colloca 20esima nel ranking di feature prodotto sul dataset *acoust-feats* e 40esima nel caso del nuovo riaddestramento su *dataset-top40* confermandosi, tra quelle selezionate, la meno rilevante.

Per quanto riguarda, invece, i valori dei punteggi prodotti a seguito di tale riaddestramento, essi non si discostano di molto dal range di valori prodotto nei singoli addestramenti per *ling-feats* e *acoust-feats*.

[15] Per maggiori approfondimenti riguardo il *RASTA-style auditory spectrum* si rimanda a Hermansky e Morgan (1994).

Molti dei risultati appena esposti coincidono, a grandi linee, con quelle che si ritrovano anche in Boggia (2021).

6.6 Paragone nel ranking prodotto da Random Forest

Una piccola riflessione si vuole qui fare in merito al ranking delle feature definito dal modello Random Forest riaddestrato con le stesse modalità viste nel paragrafo precedente (*11-fold validation* sul sub-corpus multimodale *dataset-top40*).

Nonostante il set di feature fornito ai due modelli per il riaddestramento sia lo stesso, paragonare quali feature si posizionano in cima alla classifica sulla base del peso assegnato da Random Forest e confrontarle con quelle ritenute più importanti da SVM avrebbe garantito solidità nell'identificare le caratteristiche linguistiche di una frase determinanti il fenomeno di attenzione: se 2 classificatori su 2 usano tra le top x informazioni le stesse feature per discriminare tra frasi appartenenti all'insieme positivo o negativo, significherà, con una grande probabilità, che esse siano, di fatto, informazioni di rilevanza nella definizione del fenomeno analizzato.

In specifico si andranno ad analizzare quelle che Random Forest colloca nelle prime 10 posizioni confrontandole con il grado di importanza assegnato ad esse da SVM.

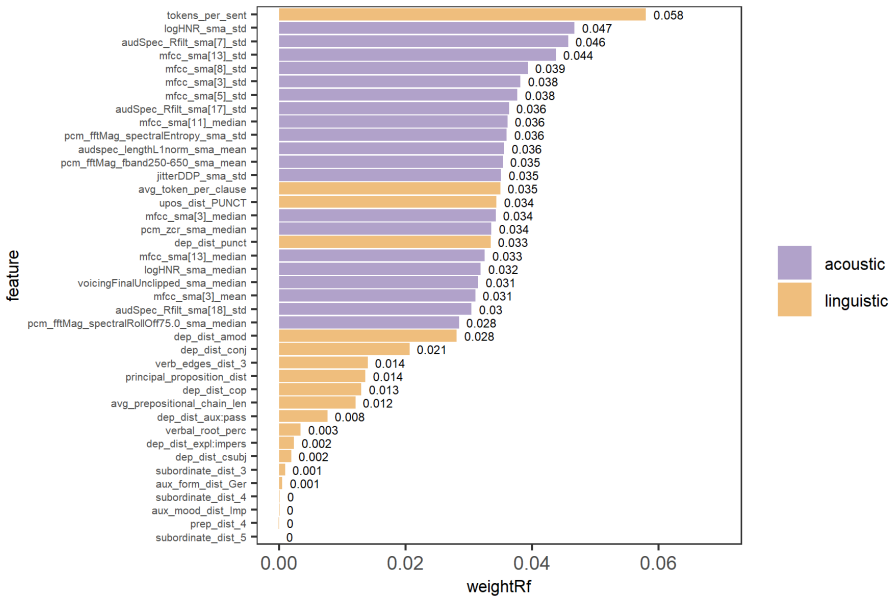


Figura 6.9: Ranking 40 feature linguistiche e acustiche prodotto da Random Forest a seguito di un riaddestramento su un sub-dataset multimodale.

Da una prima analisi del grafico di Figura 6.9 si osserva, innanzitutto, che Random Forest tende ad utilizzare molte più feature acustiche rispetto a SVM: tra le top-10, infatti, vi è solo 1 feature linguistica che si posiziona, nonostante ciò, in prima posizione. Quest'ultima (*tokens_per_sent*)[16] si ritrova in 6a posizione nel ranking prodotto da SVM, mostrandosi, dunque, come una feature che sembra contribuire in maniera abbastanza decisiva al fenomeno di attenzione.

Anche in 2a posizione si ritrova una feature ritenuta rilevante da SVM (3a posizione) che è *logHNR_sma_std*, un'informazione relativa al suono della voce.

In 3a posizione, invece, Random Forest piazza un'informazione relativa allo spettro acustico (*audSpec_Rfilt_sma[17]_std*) che SVM reputa la meno rilevante tra tutte le 40 feature fornite in input durante la fase di riaddestramento. In tal caso, dunque, non si è completamente sicuri circa la rilevanza o meno di questa informazione per la definizione del fenomeno di *attention*.

Tra le 10 feature più significative secondo Random Forest si collocano, inoltre, anche diversi coefficienti dell'MFC che si ritrovano tra le prime 20 posizioni nel caso di SVM: informazioni spettrali sembrerebbero, dunque, contribuire alla discriminazione di frasi che generano, o meno, attenzione.

Un'ultima osservazione da fare è relativa alla feature *avg_token_per_clause* che, in realtà, rientra tra le top-20 feature per Random Forest ma è la seconda feature per rilevanza per SVM: anche la lunghezza media di frase calcolata per token sembrerebbe dunque dotata di un potere predittivo abbastanza decisivo.

[16] Si tratta del numero di token per frase.

Capitolo 7

Studio sulla previsione di categorie tematiche

In questo capitolo si descriverà com'è avvenuta la fase di costruzione di un sistema di classificazione lineare volto alla predizione di informazioni tematiche. Lo studio vuole poter contribuire a trovare delle risposte (seppure derivanti dal contesto estremamente specifico del corpus di dati in possesso) a domande di carattere generale del tipo:

- Nei discorsi parlati esiste una relazione tra il piano fonetico-linguistico e semantico?
- Sulla base delle caratteristiche strutturali di una frase, è possibile prevederne il contenuto?

Per far ciò si procederà, dapprima, a processare opportunamente il dataset costruito nel capitolo 4, per poi addestrare due modelli (SVM e Random Forest) con una sola modalità (l'*11-fold cross validation*), ma fornendo in input 4 diverse tipologie di dataset. Si procederà, infine, a commentare i risultati ottenuti dagli 8 esperimenti cercando di individuare eventuali connotati di quella che, negli studi linguistici, viene chiamata *interfaccia sintattico-semantica*. Di quest'ultima si darà una definizione nel prossimo paragrafo.

7.1 Pattern sintattici e prosodico-acustici per la categorizzazione tematica del discorso

Si è visto, nei capitoli iniziali del presente lavoro, che uno degli scopi primari del progetto CHROME è quello di costruire un avatar in grado di comunicare come una guida esperta.

Una delle sfide è quella di fare in modo che l'avatar non abbia un set predefinito di frasi da cui attingere, ma che generi da sé delle frasi o, più in generale, dei discorsi che siano soddisfacenti sia in termini di contenuto ma anche in termini

comunicativi e gestuali.[1] Il progetto CHROME si può, per tali ragioni, definire un vero e proprio studio improntato sulla comunicazione a tutto tondo.

Sulla scia dei vari studi condotti presso l'Istituto di Linguistica Computazionale di Pisa, è stato inevitabile procedere alla costruzione di un classificatore di *attention*. La mia analisi, però, come più volte sottolineato, è provvista di un'informazione in più annotata manualmente durante la fase di costruzione e definizione del dataset descritta nel Capitolo 4: l'informazione relativa all'argomento tematico di ogni frase.

Nel momento in cui si è pensato di arricchire il dataset CHROME con l'informazione delle categorie tematiche si aveva in mente di condurre un'analisi riguardante il rapporto tra le caratteristiche linguistiche e prosodico-acustiche, da un lato, e quelle più prettamente "semantiche" dall'altro.

Le categorie tematiche, infatti, per quanto macroscopiche e arbitrarie possano essere, forniscono un'informazione molto differente rispetto a quelle annotate automaticamente dai tool *Profiling-UD* e *OpenSMILE*: esse riportano un'informazione di tipo semantica attinente, pertanto, al significato veicolato dal messaggio.

Nel corso del tempo, uno degli interessi centrali nel campo della linguistica è stato proprio quello di comprendere il modo in cui il piano sintattico e semantico interagissero tra di loro. In particolare, si cerca di individuare i connotati di quella che viene definita un'*interfaccia sintattico-semantica*:

> «L'interfaccia sintattico-semantica è il livello della grammatica dove la relazione tra la sintassi e la semantica si manifesta» (Sauerland e Stechow 2000)[2]

Come si è già detto, nel nostro caso, oltre a informazioni sintattiche si è in possesso di informazioni fonetico-acustiche che si possono, in un certo qual modo, accostare a quelle linguistiche in virtù del fatto che si tratta di connotati che un parlante conferisce alle proprie elaborazioni orali in maniera del tutto inconsapevole. Quando comunicano i parlanti non pongono generalmente attenzione a come le informazioni vengono trasmesse (ovvero con quali strutture sintattiche o caratteristiche prosodico-acustiche vengono costruite le frasi di un discorso parlato), ma si può presumere che le elaborino sulla base di quello che vogliono comunicare. L'ipotesi è, cioè, che a certi significati corrispondano delle strutture sintattiche e delle caratteristiche prosodico-acustiche specifiche.

Ad esempio, nel particolare caso del dataset CHROME, si potrebbe ipotizzare che la guida elabori delle frasi più lunghe quando descrive gli ambienti della Certosa o che, al contrario, usi delle frasi più corte in un altro contesto comunicativo, come, per esempio, quando risponde alle domande dei visitatori.

[1] Non ci si deve dimenticare, infatti, che tra i materiali raccolti in seno al progetto CHROME vi sono anche dei video che vengono analizzati da uno specifico team di ricerca con lo scopo di identificare gli aspetti gestuali più salienti della guida e di come questi influiscano nella trasmissione dei contenuti.

[2] «The syntax-semantics interface is the level of grammar where the relationship between syntax and semantics is established».

O ancora, si potrebbe supporre che la guida elabori delle frasi con una maggiore intensità prosodica quando si rivolge in maniera diretta al pubblico e che ponga, al contrario, minore forza intonativa quando elabora delle informazioni a carattere più esplicativo.

Si potrebbe, poi, provare a estendere queste ipotesi, originatesi da un contesto specifico, a un contesto più ampio, provando a riflettere sull'esistenza di una possibile relazione tra i piani linguistici in questione e sulla possibilità di fare previsioni circa il contenuto (la semantica) di un enunciato sulla base delle sue caratteristiche linguistico-acustiche e viceversa.

Per cercare di indagare meglio questi aspetti, l'idea è stata quella di costruire un classificatore in grado di prevedere le categorie tematiche sulla base delle caratteristiche linguistiche e fonetico-acustiche delle frasi componenti il dataset, per poi studiare il ranking delle feature da esso prodotto in maniera tale da identificare quelle maggiormente significative nel processo di assegnazione di una categoria piuttosto che un'altra. L'analisi dei risultati di performance ottenuti dal classificatore potrebbe, inoltre, dare indicazioni su un'eventuale risposta (affermativa o negativa che sia) circa le interazioni tra i piani di cui sopra si è parlato.

Ovviamente eventuali conclusioni si riferirebbero al contesto estremamente specifico coperto dal dataset di cui si è in possesso, eppure esse potrebbero parzialmente porsi all'interno del dibattito circa la possibile influenza tra gli aspetti strutturali del discorso e l'interpretazione semantica dello stesso.

7.2 Trasformazioni preliminari del dataset

Se nel caso del classificatore di attenzione, l'informazione principale risiedeva nella colonna *engagement* del dataset, nel caso di un classificatore di categorie tematiche l'informazione da prevedere sarebbe stata quella contenuta entro la colonna *categories*. A differenza dell'*attention*, la colonna in questione poteva assumere 8 valori: in tal caso si sarebbe, dunque, trattato di una classificazione multi-label e non più binaria.

Prima di procedere a qualsiasi tipo di addestramento, era necessario, oltre che normalizzare il dataset secondo le modalità descritte nella sezione 5.4.1, eseguire un encoding numerico delle lettere rappresentanti ciascuna categoria. Questo step era fondamentale in quanto i modelli SVM e Random Forest di *sklearn* non possono essere addestrati su dati non numerici. Per tale ragione si è ideato uno script (*script_ classificatore2_ encodingColonneCategorie*) in grado di assegnare dei valori numerici alle 8 categorie secondo due diverse modalità indicate in Tabella 7.1: nel secondo mapping le categorie A (storia della Certosa), B (informazioni storiche), C (informazioni biografiche) e D (informazioni sui certosini) confluiscono tutte nella categoria 1.

La scelta di effettuare il secondo mapping, oltre che fornire uno scenario di classificazione in più, è giustificata dall'osservazione per cui ognuna delle categorie sopra menzionate introduceva un'asse di analisi troppo sottile che avrebbe potuto confondere il sistema al momento della classificazione. Unirle,

category	encoding 1	encoding 2
A (storia della Certosa)	1	1
B (informazioni storiche)	2	1
C (informazioni biografiche)	3	1
D (informazioni sui certosini)	4	1
E (descrizione arte/architettura)	5	2
F (interazione)	6	3
G (meta-informazioni)	7	4
H (miscellanea)	8	5

Tabella 7.1: Mapping dei 2 encoding numerici effettuati sulla colonna *categories*.

invece, in un'unica classe (veicolante in maniera più generica delle informazioni storiche) ha costituito un modo per renderle più individuabili e differenziate dalle altre categorie veicolanti informazioni di tipo differente (es. descrizione arte/architettura, interazione, etc).

7.3 Gli scenari di classificazione

Di un secondo, possibile, scenario di classificazione da contrapporre al primo caratterizzato da una classificazione a 8 etichette (cfr. Table 7.1, colonna *encoding 1*) si è già parlato nel paragrafo precedente: esso prevede la presenza di un dataset caratterizzato dall'unione di 4 categorie (A: storia della Certosa, B: informazioni storiche, C: informazioni biografiche e D: informazioni sui certosini) tutte a carattere storico, facendo sì che la classificazione si riducesse a 5 etichette (cfr. Tabella 7.1, colonna *encoding 2*).

Un terzo e quarto scenario di classificazione si possono identificare, invece, a partire da ognuno degli scenari sopra descritti eliminando le righe del dataset appartenenti alla categoria H (miscellanea). Quest'ultima raggruppa, infatti, sotto di sé, frasi che non appartengono a nessuna delle altre 7 categorie, identificandosi, dunque, al suo interno, come una categoria estremamente eterogenea che si presume possa introdurre "rumore" all'interno del dataset. Per tale ragione sembrava opportuno provare ad analizzare le performance del classificatore nel caso in cui questo fosse addestrato senza la presenza delle 108 frasi rientranti nella categoria sopra menzionata. Una panoramica dei possibili scenari di classificazione è visibile in Tabella 7.2.

I modelli di classificazione prescelti sono stati, anche in questo caso, SVM e Random Forest. A differenza del classificatore di attenzione costruito nel capitolo 6, però, in tutti gli scenari gli algoritmi sono stati addestrati esclusivamente con una *11-fold cross validation*,[3] adattando lo script già costruito per il precedente classificatore a una classificazione multi-label.

[3] Cfr. *supra*, par. 6.3.1.

	A	B	C	D	E	F	G	H	caratteristiche
scenario 1	1	2	3	4	5	6	7	8	8 etichette, 1114 record
scenario 2	1	1	1	1	2	3	4	5	5 etichette, 1114 record
scenario 3	1	2	3	4	5	6	7	-	7 etichette, 1006 record
scenario 4	1	1	1	1	2	3	4	-	4 etichette, 1006 record

Tabella 7.2: Caratteristiche dei 4 scenari per il classificatore di categorie tematiche.

7.4 Risultati degli esperimenti

Per analizzare le performance dei classificatori addestrati, si è deciso di calcolare, oltre che il *classification report*, anche la *confusion matrix* che consente di analizzare il comportamento del classificatore durante il processo di assegnazione di ciascuna categoria tematica.[4]

La prima cosa da osservare, nel caso di questo esperimento, è che il dataset si presenta abbastanza sbilanciato nella distribuzione delle categorie come si evince dalla Tabella 7.3.

Categoria	Distribuzione
A (storia della Certosa)	62
B (informazioni storiche)	76
C (informazioni biografiche)	157
D (informazioni sui certosini)	150
E (descrizione arte/architettura)	396
F (interazione)	94
G (meta-informazioni)	71
H (miscellanea)	108
tot.	1114

Tabella 7.3: Numero di record per ciascuna delle 8 categorie tematiche.

Questo ha come diretta conseguenza l'impossibilità di prendere l'accuratezza come metrica di riferimento per la valutazione dei modelli, dal momento che essa non costituisce una metrica affidabile nel caso di dataset non bilanciati. Per tale ragione si tenderanno a commentare prevalentemente i valori di *f1-score* che, invece, essendo la media pesata di due metriche quali *precision* e *recall*, si presta bene ai fini di valutazione di un dataset quale quello in nostro possesso. Essendo, però, la *f1-score* calcolata per ogni categoria possibile, si riporteranno, di volta in volta, i valori sui quali si tenderà maggiormente a riflettere. Per una visione completa dei risultati si rimanda, comunque, all'Appendice B.1.

Osservando le prestazioni generali di tutti modelli si possono fare le seguenti considerazioni:

[4]L'elenco completo sia dei *classification report* che delle *confusion matrix*, suddivisi per modelli e scenari, si può visionare in Appendice B. Per una definizione più approfondita di *confusion matrix* si consulti, invece, il paragrafo 5.5.

- Tendenzialmente si nota che tutti modelli, in pressoché tutti gli scenari (ma in particolare negli scenari 1 e 3), tendono ad assegnare, erroneamente, la categoria E (descrizione arte/architettura) sia a record di categorie che riescono facilmente a identificare sia a quelli di categorie che tendono ad individuare con maggiore fatica. Questa tendenza è giustificata dalla distribuzione dei dati vista in Tabella 7.3: la categoria E è, infatti, la più popolata;

- Il fenomeno di cui si parlava al precedente punto tende a ridursi nei casi degli scenari 2 e 4 in quanto caratterizzati dall'unione delle categorie A, B, C e D (storia della Certosa, informazioni storiche, informazioni biografiche e informazioni sui certosini). La presenza di un'unica categoria per queste ultime tende a controbilanciare il numero di record presenti nella categoria E (descrizione arte/architettura) rendendo la distribuzione degli elementi leggermente più bilanciata (445 vs 396). Una conseguenza di ciò è che le performance dei modelli in questi scenari sono, complessivamente, superiori rispetto agli scenari 1 e 3.

Analizzando, poi, più in specifico i singoli scenari, si possono fare le seguenti osservazioni:

- Scenario 1: SVM tende spesso ad assegnare la categoria E (descrizione arte/architettura). Si registrano pertanto, valori di *f1-score* abbastanza bassi, specialmente per le categorie A (storia della Certosa) e B (informazioni storiche): rispettivamente 0.17 e 0.24. Random Forest è, invece, del tutto incapace di individuare elementi appartenenti alle categorie A e B e ha la tendenza ad assegnare la categoria più frequente (la E);

- Scenario 2: a seguito del raggruppamento delle categorie "storia della Certosa", "informazioni storiche", "informazioni biografiche" e "informazioni sui certosini" (A, B, C e D) la situazione migliora di molto. Come già detto, la presenza di un'unica categoria per le 4 sopra menzionate tende a controbilanciare il numero di elementi presenti nella categoria "descrizione arte/architettura" (E) rendendo la distribuzione degli stessi più bilanciata. In termini di *f1-score*, i valori tendono ad innalzarsi notevolmente rispetto allo scenario precedente;

- Scenario 3: lo scenario 3, pur continuando a mantenere la distinzione tra le categorie A (storia della Certosa), B (informazioni storiche), C (informazioni biografiche) e D (informazioni sui certosini), prevede la rimozione della categoria H (miscellanea) rappresentante, in un certo senso, del rumore all'interno del dataset in quanto contenente, per definizione, frasi molto diverse tra loro. Contrariamente a quanto si pensava, la rimozione di una tale tipologia di frasi non ha introdotto alcun miglioramento nelle performance del classificatore. Si registra, infatti, una situazione molto simile allo scenario 1 in cui la categoria E (descrizione arte/architettura) domina sulle altre e i valori di *f1-score* sono piuttosto bassi per le categorie A e B;

- Scenario 4: si tratta, in assoluto, dello scenario migliore dal momento che, oltre alla rimozione della categoria H (miscellanea), vengono unificate le categorie a carattere più genericamente storico. Guardando la *confusion matrix* di SVM, infatti, si nota che l'algoritmo è in grado di individuare molti *True Positive*: segno che le predizioni avvengono in maniera corretta. L'affidabilità delle predizioni positive sembra essere confermata anche dai valori di *f1-score* ottenuti da SVM in questo scenario: 0.70 per la categoria ABCD (informazioni storiche), 0.63 per la categoria E (descrizione arte/architettura), 0.57 per F (interazione) e 0.53 per G (meta-informazioni). Anche Random Forest presenta, in questo caso, dei risultati abbastanza elevati: a fronte di una precisione accettabile nell'individuazione delle categorie ABCD, E e F, non sembra tuttavia essere molto preciso nell'identificazione della categoria G (alla quale assegna correttamente soli 2 elementi).

Osservata, dunque, una maggiore affidabilità nelle predizioni prodotte dai modelli dello scenario 4, si prenderanno questi ultimi per la fase di studio delle feature e per quella di nuovo addestramento che si condurrà nel prossimo paragrafo.

7.5 Riaddestramento del classificatore su un subset di feature

Uno dei vantaggi principali di costruire e addestrare modelli di classificazione spiegabili è la possibilità di poter analizzare, a posteriori, il loro comportamento con la speranza di avere una visione più chiara e approfondita di un certo fenomeno di interesse.

Nelle sottosezioni seguenti, si procederà, dapprima, allo studio dei ranking di feature prodotti dai 2 modelli di classificazione (SVM e Random Forest), entrambi addestrati con il dataset dello scenario 4 caratterizzato, come visto nel paragrafo 7.3, dall'unificazione delle categorie A, B, C e D (storia della Certosa, informazioni storiche, informazioni biografiche e informazioni sui certosini) e dalla rimozione dei record corrispondenti alla categoria H (miscellanea).[5]

Dopo aver osservato eventuali feature comuni tra i due modelli, verrà selezionato un subset di feature sulla base del quale condurre un riaddestramento dei modelli sopra menzionati, al fine di notare un'eventuale differenza nella rilevanza assegnata da ciascuno di essi alle feature in questione.

Addestrare nuovamente i modelli su un sottoinsieme di feature, oltre che fornire tutta una serie di vantaggi di cui si è già discorso nella sezione 5.6, offre la possibilità di annotare e comprendere le relazioni presenti tra le feature selezionate: individuare e analizzare più approfonditamente simili relazioni è, si ricordi, uno degli obiettivi che ci si è prefissati nel presente studio.

[5] Il ranking completo dello scenario 4 è visionabile in Appendice B.3.1.

7.5.1 Analisi e selezione delle feature

Al fine di individuare le feature che maggiormente contribuiscono all'attribuzione di una categoria tematica piuttosto che un'altra, si è deciso di mettere a confronto le 40 migliori feature secondo SVM con le 40 migliori feature secondo Random Forest. I due set di feature (cfr. Figura 7.1 e 7.2) sono stati riordinati in ordine decrescente sulla base dei pesi prodotti rispettivamente da ciascun modello nel primo giro di addestramento condotto nello scenario 4.[6]

Lo script *script_paragona_feats.py* ha, poi, agevolato l'individuazione delle feature che entrambi i classificatori considerano rilevanti ai fini della classificazione. In particolare, sono emerse 9 feature comuni le cui caratteristiche sono riportate in Tabella 7.4.

	Group
SYNTACTIC FEATURES	
avg_token_per_clause	Global and Local Parsed Tree Structures
SPECTRAL LLD	
mfcc_sma[1]_mean	MFCC 1–14 (Cepstral)
mfcc_sma[1]_median	MFCC 1–14 (Cepstral)
mfcc_sma[6]_mean	MFCC 1–14 (Cepstral)
mfcc_sma[6]_median	MFCC 1–14 (Cepstral)
mfcc_sma[10]_std	MFCC 1–14 (Cepstral)
pcm_fftMag_spectralHarmonicity_sma_std	Psychoacoustic sharpness, harmonicity (Spectral)
pcm_fftMag_fband1000-4000_sma_std	Spectral energy 250–650 Hz, 1 k–4 kHz (Spectral)
VOICING RELATED LLD	
F0final_sma_std	F0 (Prosodic)

Tabella 7.4: Descrizione delle 9 feature comuni a SVM e Random Forest rientranti nel ranking delle top 40.

Al di là del peso che ogni classificatore associa a ogni feature, il fatto che, tra le prime 40 per ognuno di essi, occorrono delle feature condivise, è segno che esse debbano essere considerate rilevanti per l'attribuzione delle categorie tematiche a ciascuna delle frasi del corpus.

Si tratta prevalentemente di informazioni acustiche relative allo spettro: la maggior parte sono, infatti, coefficienti dell'MFC.[7] Oltre a queste vi è 1 sola informazione relativa al suono della voce (*F0final_sma_std)* e 1 sola informazione linguistica (*avg_token_per_clause*) relativa alla lunghezza media per frase calcolata in termini di numero di token.

Una volta individuate le feature condivise da entrambi i modelli, a questo punto la scelta è stata quella di selezionare le 40 migliori feature per SVM e

[6] Cfr. *supra*, Tabella 7.2.
[7] Cfr. *supra*, par. 6.5.1.

procedere, con queste ultime, a un riaddestramento di entrambi i modelli per vedere se questi avrebbero incrementato (o ridotto) il peso associato a ciascuna delle 9 feature individuate.

7.5.2 La fase di riaddestramento

Una volta identificato il sotto-insieme di feature su cui condurre un nuovo addestramento, si è proceduto alla definizione di uno script visionabile in allegato (*script_ classificatore2_ estraiDatasetTop40.py*) in grado di estrarre, dal dataset dello scenario 4, lo spazio dimensionale di interesse formato, come già accennato nel precedente paragrafo, da sole 40 dimensioni. Una volta in possesso del dataset *dataset-top-40-cat.csv*, questo è stato dato in pasto a SVM e Random Forest con un addestramento effettuato sempre mediante *11-fold cross validation*.

A seguito di questo riaddestramento, si osserva, per SVM, un riposizionamento di alcune feature in senso positivo o negativo (cfr. Figura 7.3). In altre parole, alcune feature assumono più importanza rispetto al precedente addestramento, altre, invece, perdono di rilevanza scendendo più in basso nella classifica.[8]

Le feature che si confermano tra le più significative, anche in termini di pesi assegnati dal modello, sono alcune feature spettrali come: *mfcc_ sma[1]_ median*, *mfcc_ sma[6]_ mean*, *pcm_ fftMag_ spectralHarmonicity_ sma_ std* che si posizionano tutte tra le prime 10.

Una particolare menzione meritano *pcm_ fftMag_ fband1000-4000_ sma_ std* e *mfcc_ sma[1]_ mean* che, rispetto al precedente addestramento di SVM, salgono di molto nel ranking: addirittura la *pcm_ fftMag_ fband1000-4000_ sma_ std* da 40esima viene portata alla 15esima posizione. Alla maggiore importanza assegnata da SVM, segue anche una rivalutazione da parte di Random Forest di entrambe le feature: in particolare della seconda (*mfcc_ sma[1]_ mean*), che viene posizionata dall'algoritmo in prima posizione ritenendola, dunque, la feature in assoluto più rilevante.

Da notare, inoltre, che l'unica informazione sintattica, nel secondo addestramento perde per SVM di importanza per la definizione del fenomeno di interesse, quasi a confermare che le feature più specificamente linguistiche non sono in grado di dare effettivamente conto del contenuto semantico delle frasi.

Il comportamento di Random Forest rispecchia, invece, quello che per certi versi ci si aspettava. Rispetto a SVM, Random Forest tende, infatti, ad assegnare a tutte le 9 feature, identificate nel paragrafo 7.5.1, un maggiore peso concentrandole tutte nelle prime 15 posizioni (cfr. Figura 7.4). Ciò è una riconferma del fatto che, pur fornendo a Random Forest un set di feature differente rispetto al precedente addestramento, esso continua a ritenere le 9 feature comuni a entrambi gli addestramenti quelle in assoluto più significanti e discriminanti il fenomeno che gli è stato richiesto di classificare.

[8] Il ranking completo di questo scenario è consultabile in Appendice B.3.2.

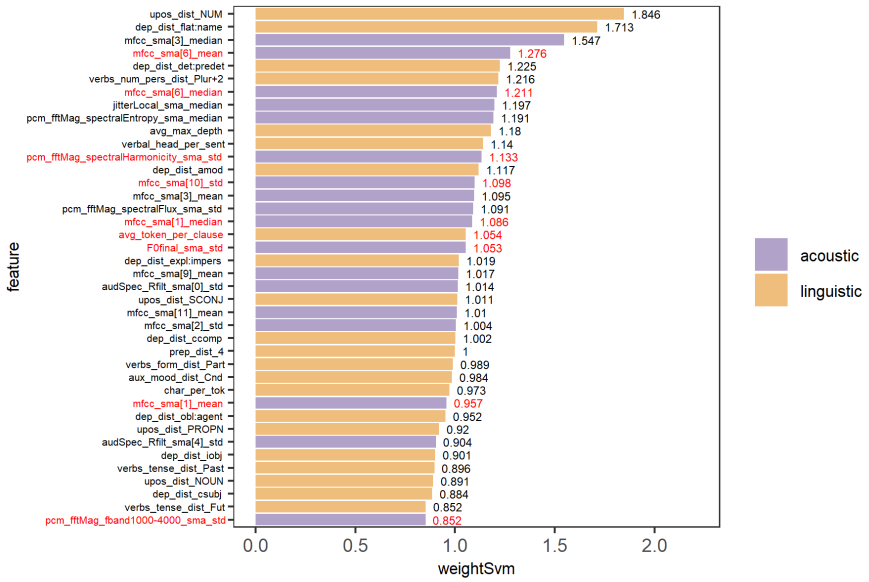


Figura 7.1: La 40 feature più significative per SVM. In rosso le feature comuni con il ranking prodotto da Random forest (vedi Figura 7.2).

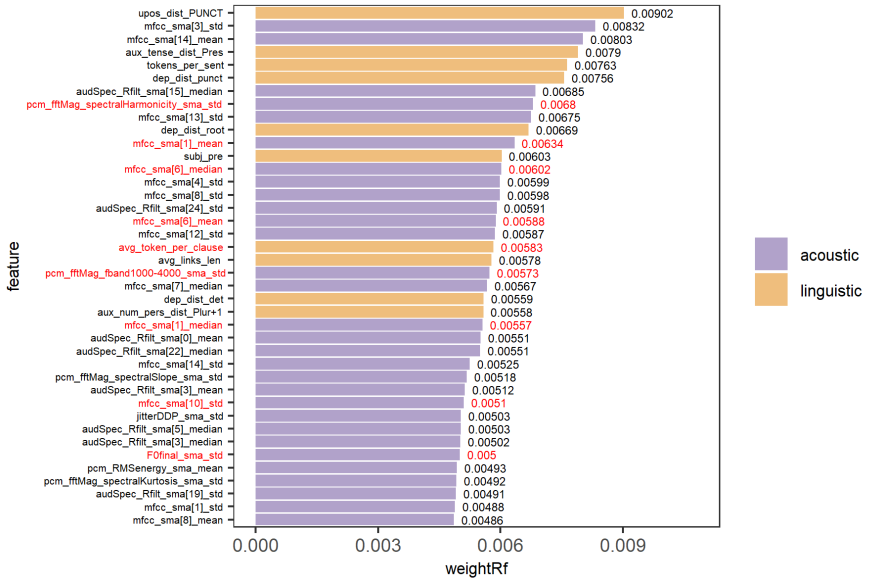


Figura 7.2: La 40 feature più significative per Random Forest. In rosso le feature comuni con il ranking prodotto da SVM (vedi Figura 7.1).

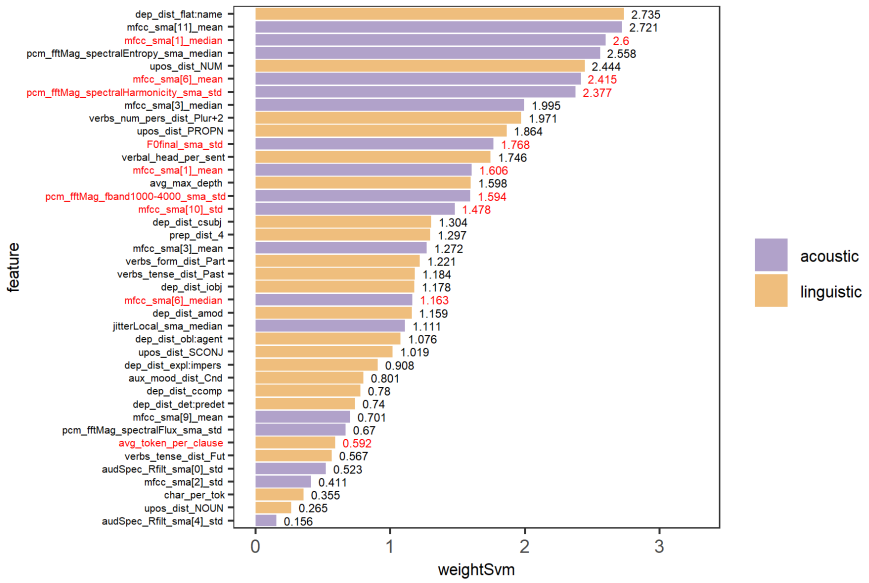


Figura 7.3: Ranking di feature prodotto da SVM a seguito del riaddestramento. In rosso le feature comuni tra SVM e Random Forest identificate nel paragrafo 7.5.1.

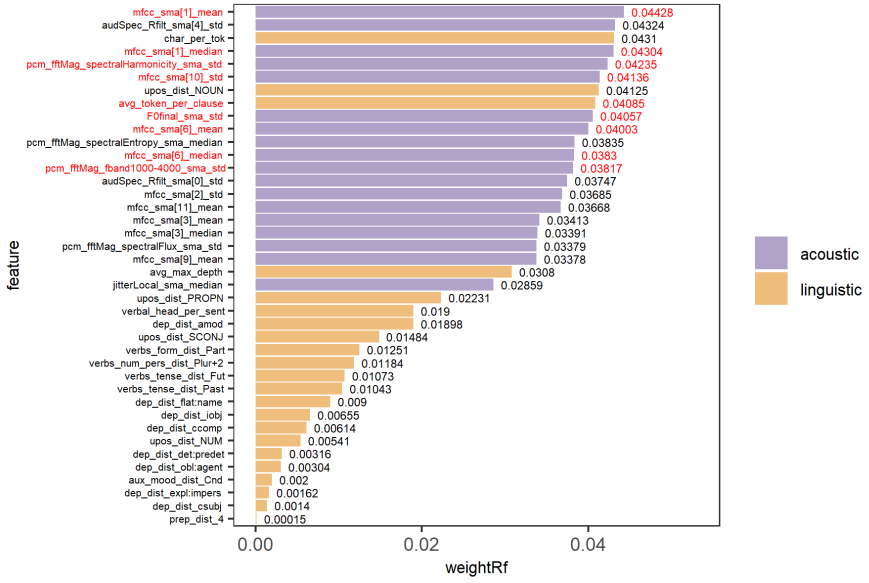


Figura 7.4: Ranking di feature prodotto da Random Forest a seguito del riaddestramento. In rosso le feature comuni tra SVM e Random Forest identificate nel paragrafo 7.5.1.

Capitolo 8

Conclusioni

In questa Tesi è stato presentato uno studio sulla comunicazione che ha previsto l'addestramento di diversi modelli di classificazione al fine di costruire, essenzialmente, due sistemi di cui indagare il comportamento: un classificatore di attenzione e un classificatore di categorie tematiche.[1]

L'interesse per questo studio nasce dalla consapevolezza di avere tra le mani una fonte composita di dati come poche ce ne sono in circolazione: raccolti a seguito di diversi anni di lavoro, i dati CHROME raggruppano, infatti, oltre che videoregistrazioni di guide turistiche e ricostruzioni di ambienti di 3 Certose del territorio campano, testi e audio di lingua parlata.

Un corpus multimodale di tale portata è una grande ricchezza, non solo perché, come visto nel corso della trattazione, le sue applicazioni sono molteplici e variegate, ma anche perché esso consente di studiare una componente fondamentale della lingua: quella orale. Il dataset costruito a partire dal corpus di CHROME consente, infatti, di studiare questa varietà diamesica della lingua in quanto contiene, al suo interno, segmenti audio con la relativa trascrizione, dando la possibilità di annotare i singoli elementi[2] con informazioni non solo linguistiche ma proprie della lingua parlata: in altre parole, informazioni fonetico-acustiche. Oltretutto, si tratta di una tipologia di dataset che, come visto nel capitolo 4, necessita di una lunga serie di manipolazioni e lavori manuali per essere "pronto all'uso", ovvero per essere utilizzato per l'addestramento di sistemi automatici per la classificazione.

Consapevole di ciò, l'Istituto di Linguistica Computazionale "Antonio Zampolli" di Pisa si è fatto promotore, in questi anni, di diversi studi volti a indagare i "comportamenti linguistici" tipici del parlato di una guida turistica.

Conducendo due studi paralleli, uno sull'attenzione e uno sulla predizione di categorie tematiche, l'obiettivo, in questo caso, è stato quello di provare a indagare il funzionamento della comunicazione in generale. Questi aspetti, apparentemente slegati tra loro, hanno in realtà due cose in comune: il contesto

[1] Cfr. *supra*, par. 7.1.

[2] Come visto nella sezione 4.2.1, ogni elemento del dataset corrisponde a una "frase" del parlato.

in cui si applicano (quello della comunicazione) e la possibilità di poterli studiare a partire dalla stessa tipologia di dati (ovvero dati multimodali annotati con informazioni e linguistiche e acustiche).

L'idea di fondo che accomuna i due studi condotti in questa trattazione è l'idea per cui alcuni aspetti strutturali del discorso possano influenzare fenomeni "esterni", ovvero fenomeni che hanno più strettamente a che fare con la comunicazione in generale: ad esempio la possibilità di generare attenzione sulla base di una serie di caratteristiche del discorso o di utilizzare un pattern ricorrente di caratteristiche sulla base del tipo di contenuto veicolato.

Per poter individuare simili interrelazioni l'idea è stata quella di annotare il corpus di dati con feature linguistiche e acustiche consolidate nella letteratura[3] e che si è dimostrato essere applicabili a una vasta molteplicità di task. Le feature linguistiche estratte dal tool *Profiling-UD*[4] possono, ad esempio, essere efficacemente usate per l'analisi dei registri linguistici, oppure nel campo della sociolinguistica per individuare differenze diastratiche nell'uso della lingua, o ancora nel campo della stilometria per l'attribuzione di un autore a un testo sulla base delle caratteristiche stilistiche dello stesso.[5] Anche *OpenSMILE* (il tool usato per estrarre le feature acustiche)[6] è stato pensato dai suoi ideatori per fornire un set di feature acustiche utilizzabili in una varietà di campi. Task ben noti entro la comunità scientifica interessata a studiare il segnale acustico sono, ad esempio, il rilevamento di discorsi ingannevoli, del grado di sincerità di un parlante o, ancora, del suo essere più o meno nativo di una determinata lingua.[7] Tutto questo per dire che informazioni di tal genere forniscono un buon punto di partenza per lo studio e l'analisi di un qualsivoglia fenomeno linguistico a cui si è interessati.

Annotato, perciò, il dataset con queste informazioni, si è potuto procedere all'addestramento di sistemi di classificazione al fine di osservare l'utilizzo che questi avrebbero fatto delle informazioni fornite in ingresso per prevedere i fenomeni oggetto di studio. Come si è spiegato in 5.1, infatti, sistemi di apprendimento automatico vengono solitamente utilizzati in tutte quelle occasioni in cui si vorrebbe avere una conoscenza più approfondita di un dato fenomeno che, però, non si riesce né a spiegare né a formalizzare entro un modello teorico preciso.

Ciò che è emerso, nel primo studio, è che tra le feature più significative per classificare l'attenzione vi sarebbero alcune informazioni acustiche (sia spettrali che relative alla tonalità e all'energia della voce) e diverse informazioni relative alla struttura sintattica (tra cui spicca la lunghezza media per frasi calcolata in termini di token, ovvero l'*avg_token_per_clause*).

[3] Cfr. Brunato et al. (2020) per le feature linguistiche e Eyben, Wöllmer e Schuller (2010) per quelle acustiche.

[4] Cfr. *supra*, par. 4.2.3.

[5] Cfr. *supra*, sottosezione 4.2.3.1 e Brunato et al. (2020).

[6] Cfr. *supra*, par. 4.2.5.

[7] Questi task sono quelli che Schuller e colleghi definiscono *deception*, *sincerity* e *native language sub-challenges* (Schuller et al. 2016).

A definire, invece, nel secondo studio, i connotati di una possibile relazione esistente tra gli aspetti fonetico-sintattici e semantici del discorso vi sarebbero alcune feature spettrali e una feature linguistica (la stessa *avg_ token_ per_ clause* menzionata poco sopra).

I risultati ottenuti nei due studi (e che sono stati qui sintetizzati) tengono conto delle feature più "significative". Per ritenere una feature significativa nella descrizione di un fenomeno si è guardato non solo alla sua eventuale ricorrenza nei ranking dei vari addestramenti condotti ma anche alla posizione assunta dalla feature all'interno dello stesso ranking prodotto dal modello.

A questo punto, però, sorge spontanea una domanda: si potrebbero identificare delle feature linguistiche o acustiche che permettono di descrivere entrambi i fenomeni oggetto di studio? Feature che, in altre parole, si presenterebbero più significative di altre per la descrizione di certi fenomeni comunicativi?

A conclusione di questo lavoro di Tesi, l'idea sarebbe quella di individuare, all'interno di due set di feature già di per sé significativi per la descrizione di certi fenomeni linguistici (quelli forniti da *Profiling-UD* e *OpenSMILE*) un sottoinsieme di feature ancora più distintivo ed efficace volto all'identificazione di alcuni eventi comunicativi.

Il confronto delle eventuali feature comuni ai due studi può essere fatto sulla base dei ranking di feature prodotti per ognuno degli studi dal modello SVM sul dataset *all-feats* (contenente sia feature linguistiche che acustiche, ma anche informazioni categoriali).

Mettendo a confronto i due ranking in questione (con l'ausilio dello stesso script *script_ paragona_ feats.py* usato nella sezione 7.5.1) si osserva che vi sono 10 feature comuni, tra cui:

- 2 informazioni relative alla struttura sintattica (*avg_ token_ per_ clause* e *prep_ dist_ 4*)
- 4 feature veicolanti informazioni di tipo sintattico (del tipo *dep_ dist_ **);
- 3 coefficienti dell'MFC, dunque informazioni relative allo spettro acustico (del tipo *mfcc_ sma**).
- 1 informazione spettrale *RASTA-style* del tipo *audSpec_ Rfilt_ sma**;

Analizzando questo insieme di feature comuni, si nota che molte di esse, indipendentemente dalla posizione assunta, si ritrovano nei ranking analizzati nelle sezioni 6.4 e 7.4, o comunque, in essi, è possibile trovare feature appartenenti, per così dire, alla stessa "tipologia".

Seppure non menzionati nelle analisi fatte, ad esempio, molti coefficienti dell'MFC vengono utilizzati da SVM per discernere tra frasi che genereranno o non genereranno attenzione. Oppure le informazione sintattiche del tipo *dep_ dist_ ** rientrano di fatto tra le prime 40 feature più importanti per entrambi gli studi (si confrontino le Figure 6.8 e 7.3). Lo stesso vale per l'informazione spettrale del tipo *audSpec_ Rfilt_ sma**. Le 2 feature relative alla struttura sintattica sono, invece, le uniche per le quali non si deve fare un discorso di appartenenza a un

gruppo: in entrambi gli studi si ritrovano precisamente *avg_ token_ per_ clause* e *prep_ dist_ 4*.

Ad ogni modo, l'organizzazione sintattica del discorso, le relazioni sintattiche che si instaurano tra i componenti del discorso e le informazioni relative allo spettro acustico sembrerebbero informazioni alle quali un parlante conferisce, più o meno consapevolmente, una certa importanza e/o un certo significato a livello comunicativo.

Ovviamente le conclusioni qui fatte non hanno la pretesa di avere un carattere di assolutezza e universalità. Al contrario si è ben consapevoli dei limiti sia dei dati che delle metodologie utilizzate per condurre questo studio.

Innanzitutto, la natura (nonché la quantità) dei materiali è limitata a un contesto estremamente specifico: quello dei beni culturali. In particolare, i dati riproducono la comunicazione di una guida turistica che descrive gli ambienti della Certosa di San Martino di Napoli. Questo fatto potrebbe influire sulla possibilità di generalizzare i risultati ottenuti in quanto derivanti dalle osservazioni fatte su una sola guida turistica in una situazione comunicativa circoscritta.

Un ulteriore limite degli studi di cui si è parlato è la selezione fatta in merito alle "modalità" dei dati che si è deciso di utilizzare. Per rendere l'analisi più semplice è stata eliminata una componente essenziale della comunicazione verbale tra umani, pur presente originariamente nei dati CHROME: quella gestuale. Proprio per la sua importanza, sarebbe interessante in futuro arricchire lo studio di questa componente che è già stata oggetto di indagine da parte di alcuni studiosi come ad esempio Cataldo et al. (2019).

Altro aspetto da considerare è la scelta di non essersi occupati del "contenuto" delle frasi del corpus ma di essersi concentrati esclusivamente sulla "forma": non ci si è chiesto, ad esempio, quali siano le parole che la guida usa per generare attenzione o per veicolare un certo contenuto, limitandosi a osservare le caratteristiche linguistiche e fonetico-acustiche più "macroscopiche" di una frase.[8] Anche questo potrebbe essere un'ulteriore spunto di riflessione per chi in futuro voglia proseguire nella direzione presa da questo studio.

Un altro punto sul quale ci si potrebbe concentrare in un'eventuale ripresa e prosecuzione delle indagini riguarda più specificamente la costruzione e l'addestramento del classificatore di categorie tematiche. L'invito è quello di ampliare ed esplorare più in dettaglio il campo delle opzioni possibili fornendo ai modelli varie combinazioni di dati e feature (al pari di quanto è stato fatto con il classificatore di attenzione). Il numero limitato di esperimenti condotti è giustificato dall'esiguo tempo di cui si è potuto disporre per mettere in piedi l'esperimento.

Infine, si è consapevoli della parziale arbitrarietà con cui i fenomeni oggetto di interesse siano stati fatti rientrare all'interno di quello che è stato definito uno "studio sulla comunicazione". Riguardo questo aspetto si vuole qui fornire una spiegazione ulteriore.

Si è già illustrato in più punti l'obiettivo primario del progetto CHROME: costruire agenti artificiali in grado di comunicare in maniera efficace. Si è anche

[8]Da sottolineare, però, che nel secondo studio la scelta di "fare semantica" senza la "semantica" è stata voluta. La sfida è stata, infatti, quella di prevedere il significato (seppur generico e macroscopico) delle frasi a partire da informazioni morfosintattiche e acustiche.

detto, però, che le macchine non solo non conoscono le "buone maniere" della comunicazione tra umani, ma non possiedono ancora nemmeno la capacità di valutare le reazioni dei loro interlocutori e di adattarsi di conseguenza alle diverse situazioni comunicative (che cambiano continuamente a seconda degli interlocutori e dello specifico contesto entro cui ci si trova).

L'unica via percorribile per fare in modo che un utente umano abbia un'esperienza positiva nell'utilizzo di questi sistemi è renderli in grado di comunicare efficacemente in qualsiasi situazione: dotarli, in altre parole, di una strategia comunicativa, per così dire, "universale", valida in qualsiasi contesto.

Come già spiegato nell'Introduzione (capitolo 1), l'individuazione degli aspetti del discorso in grado di generare attenzione negli interlocutori con i quali si comunica sarebbe di grande aiuto per il raggiungimento di tale scopo e lo stesso si può dire per l'individuazione di una possibile relazione tra il piano sintattico e fonetico-acustico del discorso e il modo in cui questo viene interpretato dai parlanti. Se è vero che esistono delle differenze strutturali a seconda di quale sia il contenuto veicolato, l'individuazione di una possibile dipendenza tra il piano strutturale e il piano semantico obbedisce alla stessa esigenza di trovare una modalità di trasmissione dei contenuti valida e persuasiva.

Studi come questo, che mirano a indagare quali siano gli aspetti fondanti della lingua parlata ai quali gli umani conferiscono più importanza, potrebbero non solo di gran lunga migliorare e contribuire alla costruzione di agenti artificiali più efficaci a livello comunicativo, ma potrebbero contribuire ad accrescere la conoscenza che gli umani stessi possiedono circa la natura e i principi che sottostanno il tipo di comunicazione da loro utilizzato: il linguaggio verbale.

Questo studio costituisce, ovviamente, una prima esplorazione di un campo tanto affascinante quanto di interesse secolare e di cui oggi si è in grado di fornire aspetti differenti rispetto al passato. Se Cicerone parlava di "discorsi eleganti, di concetti ed espressioni appropriate e di controllo e modulazione dalla voce" (si veda l'Introduzione, capitolo 1), oggi, grazie all'ausilio del *Machine Learning*, si può determinare in maniera più precisa quale sia quell'espressione appropriata o quella peculiare caratteristica della voce che renderà il discorso perfetto e l'interlocutore dilettato ed emotivamente coinvolto.

Per quanto, però, l'individuazione degli aspetti dell'elocuzione e della declamazione affascini da sempre l'uomo, non ci si dimentichi che comunque la comunicazione, specie quella umana, è un fenomeno estremamente complesso all'interno del quale entrano in gioco numerosissimi fattori a tal punto che sarebbe molto difficile giungere alla definizione di quella strategia comunicativa universale di cui sopra si parlava.

Raggiungere il prototipo di comunicazione "ideale" è certamente uno degli obiettivi e delle speranze di progetti come questo. Tale scopo potrebbe anche essere raggiunto e uno dei contesti di applicazione potrebbe essere, come visto, quello artificiale. Tuttavia è proprio questa idealità, questa perfezione, che renderebbe le macchine sempre e solo delle semplici "macchine".

A proposito del famoso gioco dell'imitazione (di cui pure si è parlato nell'Introduzione, capitolo 1), Turing scrive:

> «Colui che interroga potrebbe distinguere la macchina dall'uomo semplicemente ponendo ad entrami un certo numero di problemi aritmetici. La macchina verrebbe smascherata per la sua tremenda precisione. [...] La macchina (programmata per giocare il gioco) non cercherebbe di dare la risposta esatta a problemi aritmetici. Introdurrebbe deliberatamente degli errori, in un modo studiato apposta per confondere chi interroga» (Turing 1950)[9]

Se l'obiettivo è quello di rendere i sistemi artificiali il più possibile simili a noi, essi dovranno imitarci in tutto e per tutto rendendo, come ci insegna un altro grande poeta latino Terenzio, «niente di umano estraneo a loro» (Terenzio, *Heautontimorumenos*)[10]: nemmeno un'eventuale imperfezione nel modo di comunicare.

[9] «It is claimed that the interrogator could distinguish the machine from the man simply by setting them a number of problems in arithmetic. The machine would be unmasked because of its deadly accuracy. [...] The machine (programmed for playing the game) would not attempt to give the right answers to the arithmetic problems. It would deliberately introduce mistakes in a manner calculated to confuse the interrogator».

[10] «Homo sum, humani nihil a me alienum puto», Terenzio (2016, a cura di Lisa Piazzi).

Bibliografia

Altimari, Veronica (2020). "Covid 19, con l'intelligenza artificiale risposte in venti secondi". In: *Roma Today*. URL: https://www.romatoday.it/attualita/coronavirus-diagnosi-covid-19-intelligenza-artificiale.html (visitato il 14/11/2021).

Ansani, Alessandro (2019). "The meanings of the sigh. Vocal expression along the route of our desires". In: *Lebenswelt. Aesthetics and philosophy of experience*.

Boggia, Federico (2021). "Multimodalità del coinvolgimento: creazione di un dataset ed esperimenti sull'engagement tra guida e visitatori in siti di interesse culturale". Tesi di Laurea Triennale in Informatica Umanistica. Università di Pisa.

Bousquet, Olivier, Stéphane Boucheron e Gábor Lugosi (2004). "Introduction to Statistical Learning Theory". In: *Advanced Lectures on Machine Learning*. A cura di Olivier Bousquet, Ulrike von Luxburg e Gunnar Rätsch. Vol. 3176. Berlin, Heidelberg: Springer Berlin Heidelberg.

Brunato, Dominique et al. (2020). "Profiling-UD: a Tool for Linguistic Profiling of Texts". In: *Proceedings of the 12th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, pp. 7145–7151.

Cataldo, Violetta et al. (2019). "Phonetic and functional features of pauses, and concurrent gestures, in tourist guides' speech". In: *Gli archivi sonori al crocevia tra scienze fonetiche, informatica umanistica e patrimonio digitale*. (Visitato il 27/08/2021).

Cepstrum (2014). In: *Wikipedia*. URL: https://it.wikipedia.org/w/index.php?title=Cepstrum&oldid=68983736 (visitato il 10/10/2021).

Cera, Valeria (2020). "Semantics and Architecture: Reflections and Method Proposal for the Recognition of Semanically-Defined Architectural Forms". In: *Impact of Industry 4.0 on Architecture and Cultural Heritage*.

Certosa di San Martino (2021). In: *Wikipedia*. URL: https://it.wikipedia.org/w/index.php?title=Certosa_di_San_Martino&oldid=122813657 (visitato il 05/09/2021).

Chang, Chih-Chung e Chih-Jen Lin (2011). "LIBSVM: A library for support vector machines". In: *ACM Transactions on Intelligent Systems and Technology* 2.3.

Cicerone, Marco Tullio (2007). *L'arte di comunicare*. A cura di Paolo Marisch. Milano: Oscar Mondadori.

Cutugno, Francesco et al. (2018). "The CHROME Manifesto: integrating multimodal data into Cultural Heritage Resources". In: *Proceedings of the Fifth Italian Conference on Computational Linguistics CLiC-it 2018*. Accademia University Press, pp. 155–159.

Eyben, Florian, Martin Wöllmer e Björn Schuller (2010). "Opensmile: the munich versatile and fast open-source audio feature extractor". In: *Proceedings of the international conference on Multimedia - MM '10*. Firenze: ACM Press.

Fredricks, Jennifer A., Phyllis C. Blumenfeld e Alison H. Paris (2004). "School Engagement: Potential of the Concept, State of the Evidence". In: *Review of Educational Research* 74.1, pp. 59–109.

Gagliardi, Gloria (2018). "Inter-Annotator Agreement in linguistica: una rassegna critica". In: Proceedings of the Fifth Italian Conference on Computational Linguistics (CLiC-it 2018).

Goldberg, Patricia et al. (2019). "Attentive or Not? Toward a Machine Learning Approach to Assessing Students' Visible Engagement in Classroom Instruction". In: *Educational Psychology Review* 33.1, pp. 27–49.

Guyon, Isabelle e André Elisseeff (2003). "An Introduction to Variable and Feature Selection". In: *Journal of Machine Learning Research* 3.Mar.

Hermansky, H. e N. Morgan (1994). "RASTA processing of speech". In: *IEEE Transactions on Speech and Audio Processing* 2.4. ISSN: 1063-6676.

Ippolito, Pier Paolo (2019). *Support Vector Machines*. URL: `https://pierpaolo-28.github.io/blog/blog6/`.

Izre'el, Shlomo et al. (2020). "In search of a basic unit of spoken language". In: *In Search of Basic Units of Spoken Language. A corpus-driven approach*. John Benjamins.

Logan, Beth (nov. 2000). "Mel Frequency Cepstral Coefficients for Music Modeling". In: *Proc. 1st Int. Symposium Music Information Retrieval*.

Melhart, David, Antonios Liapis e Georgios N Yannakakis (2019). "PAGAN: Video Affect Annotation Made Easy". In: Proceedings of 8th International Conference on Affective Computing Intelligent Interaction (ACII 2019).

Multimodale (2021). In: *Wikipedia*. URL: `https://it.wikipedia.org/w/index.php?title=Multimodale&oldid=114342542` (visitato il 24/08/2021).

Origlia, Antonio et al. (2019). "Human, All Too Human: Towards a Disfluent Virtual Tourist Guide". In: *UMAP'19 Adjunct: Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*, pp. 393–399.

Poggianti, Luca (2020). "Costruzione di un corpus multimodale per lo studio delle strategie di coinvolgimento di guide turistiche in siti culturali". Tesi di Laurea Triennale in Informatica Umanistica. Università di Pisa.

Ramshaw, Lance A. e Mitchell P. Marcus (mag. 1995). "Text Chunking using Transformation-Based Learning". In: URL: `http://arxiv.org/abs/cmp-lg/9505040`.

Ravelli, Andrea Amelio, Antonio Origlia e Felice Dell'Orletta (2020). "Exploring attention in multimodal corpus of guided tours". In: Proceedings of the Seventh Italian Conference on Computational Linguistics. Bologna.

Rich, Charles et al. (2010). "Recognizing Engagement in Human-Robot Interaction". In: International Conference on Human-Robot Interaction.

Rossi, Francesca (2019). *Il confine del futuro. Possiamo fidarci dell'intelligenza artificiale?* Milano: Feltrinelli.

Sauerland, Uli e Arnim Von Stechow (2000). "The syntax-semantics interface". In: p. 16.

Savage-Rumbaugh, E. Sue e Roger Lewin (1994). *Kanzi : the ape at the brink of the human mind*. New York : Wiley. URL: `http://archive.org/details/kanzi00sues`.

Savy, Renata (2006). *Specifiche per la trascrizione ortografica annotata dei testi raccolti*.

Schettino, Loredana e Violetta Cataldo (2019). "Phonetic and functional features of lexicalized pauses in Italian". In: Proceedings of the 10th International Conference of Experimental Linguistics (ExLing 2019).

Schuller, Björn et al. (2016). "The INTERSPEECH 2016 Computational Paralinguistics Challenge: Deception, Sincerity & Native Language". In: *Interspeech 2016*. ISCA. URL: `https://www.isca-speech.org/archive/interspeech_2016/schuller16_interspeech.html`.

Sidner, Candace L. et al. (2005). "Explorations in engagement for humans and robots". In: *Artificial Intelligence* 166.1, pp. 140–164.

Sorgente, Antonio et al. (2017). "A Framework for Creating Cultural Interactive Guides". In: *AI*CH 2017*.

Straka, Milan, Jan Hajič e Jana Straková (2016). "UDPipe: Trainable Pipeline for Processing CoNLL-U Files Performing Tokenization, Morphological Analysis, POS Tagging and Parsing". In: *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*. Portorož, Slovenia: European Language Resources Association (ELRA).

Tan, Pang-Ning, Michael Steinbach e Vipin Kumar (2014). *Introduction to Data Mining*. Pearson.

Terenzio, P. Afro (2016). *Adelphoe-Heautontimorumenos*. A cura di Lisa Piazzi. Mondadori.

Turing, Alan Mathison (1950). "Computing Machinery and Intelligence". In: *Mind*.

Vicuna, Laura (8 feb. 2017). "Educational Games Design: Creating an Effec- tive and Engaging Learning Experience". Bachelor's Thesis. Helsinki Metropolia University of Applied Sciences.

Weninger, Felix et al. (2013). "On the Acoustics of Emotion in Audio: What Speech, Music, and Sound have in Common". In: *Frontiers in Psychology* 4.

Appendice A

Classificatore di attenzione

A.1 *Classification report*

A.1.1 Dataset *all-feats*

SVM report

	precision	recall	f1-score	support
engagement	0.63	0.59	0.61	569
no_engagement	0.59	0.63	0.61	531
accuracy			0.61	1100
macro avg	0.61	0.61	0.61	1100
weighted avg	0.61	0.61	0.61	1100

RANDOM FOREST report

	precision	recall	f1-score	support
engagement	0.61	0.66	0.63	569
no_engagement	0.60	0.54	0.57	531
accuracy			0.60	1100
macro avg	0.60	0.60	0.60	1100
weighted avg	0.60	0.60	0.60	1100

A.1.2 Dataset *ling-feats*

SVM report

	precision	recall	f1-score	support
engagement	0.58	0.56	0.57	566
no_engagement	0.55	0.57	0.56	534
accuracy			0.57	1100
macro avg	0.57	0.57	0.57	1100
weighted avg	0.57	0.57	0.57	1100

RANDOM FOREST report

	precision	recall	f1-score	support
engagement	0.60	0.60	0.60	566
no_engagement	0.57	0.57	0.57	534
accuracy			0.59	1100
macro avg	0.59	0.59	0.59	1100
weighted avg	0.59	0.59	0.59	1100

A.1.3 Dataset *ling-feats-withCat*

SVM report

	precision	recall	f1-score	support
engagement	0.59	0.58	0.58	563
no_engagement	0.57	0.57	0.57	537
accuracy			0.58	1100
macro avg	0.58	0.58	0.58	1100
weighted avg	0.58	0.58	0.58	1100

RANDOM FOREST report

	precision	recall	f1-score	support
engagement	0.61	0.61	0.61	563
no_engagement	0.59	0.60	0.60	537
accuracy			0.60	1100
macro avg	0.60	0.60	0.60	1100
weighted avg	0.60	0.60	0.60	1100

A.1.4 Dataset *acoust-feats*

SVM report

	precision	recall	f1-score	support
engagement	0.60	0.50	0.54	565
no_engagement	0.55	0.64	0.59	535
accuracy			0.57	1100
macro avg	0.57	0.57	0.57	1100
weighted avg	0.57	0.57	0.57	1100

RANDOM FOREST report

	precision	recall	f1-score	support
engagement	0.59	0.65	0.62	565
no_engagement	0.59	0.53	0.56	535
accuracy			0.59	1100
macro avg	0.59	0.59	0.59	1100
weighted avg	0.59	0.59	0.59	1100

A.1.5 Dataset *acoust-feats-withCat*

SVM report

	precision	recall	f1-score	support
engagement	0.59	0.53	0.56	568
no_engagement	0.55	0.61	0.58	532
accuracy			0.57	1100
macro avg	0.57	0.57	0.57	1100
weighted avg	0.57	0.57	0.57	1100

RANDOM FOREST report

	precision	recall	f1-score	support
engagement	0.60	0.67	0.64	568
no_engagement	0.60	0.53	0.56	532
accuracy			0.60	1100
macro avg	0.60	0.60	0.60	1100
weighted avg	0.60	0.60	0.60	1100

A.1.6 Dataset *categories*

SVM report

	precision	recall	f1-score	support
engagement	0.52	0.37	0.43	565
no_engagement	0.49	0.64	0.56	535
accuracy			0.50	1100
macro avg	0.51	0.51	0.49	1100
weighted avg	0.51	0.50	0.49	1100

RANDOM FOREST report

	precision	recall	f1-score	support
engagement	0.53	0.37	0.43	565
no_engagement	0.49	0.65	0.56	535
accuracy			0.51	1100
macro avg	0.51	0.51	0.50	1100
weighted avg	0.51	0.51	0.50	1100

A.2 Ranking di feature

Nelle seguenti sottosezioni si riportano i ranking dei dataset maggiormente utilizzati per la fase di analisi. I ranking di tutti gli esperimenti sono consultabili in allegato alla presente Tesi.

A.2.1 Dataset *ling-feats*

Pesi prodotti a seguito dell'addestramento con una *11-fold-cross-validation* di SVM e RandomForest sul dataset *ling-feats* (contenente solo feature linguistiche). I pesi sono stati riordinati in ordine decrescente secondo i valori assegnati da SVM.

feature	weightSvm	weightRf
avg_token_per_clause	1,83698836386941	0,023102061323104
dep_dist_conj	1,32204823985663	0,013140022257465
dep_dist_aux:pass	1,31747874155527	0,005397629453597
verbal_root_perc	1,1344796763164	0,003056564607012
avg_prepositional_chain_len	1,04334425733667	0,005721712720477
aux_mood_dist_Imp	1	6,68293145254951E-05
subordinate_dist_5	1	0
dep_dist_cop	0,987004206936781	0,01082643211864
aux_form_dist_Ger	0,979950594828174	9,66470587915934E-05
subordinate_dist_4	0,958357021529813	3,16334138065627E-05
dep_dist_amod	0,950738087752889	0,017069946730515
upos_dist_PUNCT	0,893024261474253	0,024665730068912
dep_dist_punct	0,893024261474253	0,026350644338716
principal_proposition_dist	0,889943808123689	0,007002836247172
dep_dist_csubj	0,860488471311949	0,000929674692249
verb_edges_dist_3	0,820691195957124	0,009451105386471
prep_dist_4	0,797309744504711	0,000102840024117
tokens_per_sent	0,795550606149305	0,024109995041838
dep_dist_expl:impers	0,765271445613617	0,001411358193376
subordinate_dist_3	0,764886751245212	0,000603554183849

feature	weightSvm	weightRf
dep_dist_compound	0,760099004273231	0,000668014691025
verb_edges_dist_2	0,734940819828992	0,009358784056175
prep_dist_3	0,728217671103964	0,000762211192035
avg_verb_edges	0,721754778515873	0,015918236880604
dep_dist_iobj	0,712761632625123	0,004788149665267
upos_dist_VERB	0,705539854119664	0,017328779250684
upos_dist_PRON	0,698003805675356	0,018904206626048
dep_dist_advmod	0,680548978946433	0,018713573515179
dep_dist_mark	0,675046824808753	0,011827054763987
verbs_mood_dist_Ind	0,674039773628806	0,003898997751032
dep_dist_fixed	0,659472792445696	0,004517604936248
prep_dist_2	0,65279562652583	0,003860891007737
subj_post	0,639669086222398	0,005567250664112
n_prepositional_chains	0,626875043787841	0,005668345662452
verbs_tense_dist_Past	0,620933604733366	0,005424445581428
upos_dist_AUX	0,60881661428094	0,013827503459688
aux_form_dist_Part	0,593067999885367	0,000943715002554
dep_dist_discourse	0,590603616813294	0,000482992484188
dep_dist_nmod	0,58771630064706	0,014750570273427
dep_dist_nsubj	0,574417030509096	0,018364238906668
subordinate_pre	0,537994206059928	0,005411522073883
aux_num_pers_dist_Plur+1	0,534801659845492	0,003522640329087
verb_edges_dist_1	0,517582094804155	0,004449563142362
verb_edges_dist_6	0,511045630868153	0,003355474640741
dep_dist_ccomp	0,508595265570058	0,004471627808992
aux_num_pers_dist_Sing+2	0,5	0,000188479605652
verb_edges_dist_4	0,462054023682128	0,005876398718874
dep_dist_nsubj:pass	0,459951965339126	0,003736048724801
lexical_density	0,45520366978451	0,025477537190928
upos_dist_NUM	0,452041363085021	0,003465128709904
aux_num_pers_dist_Sing+1	0,451368517646394	0,001342655727371
dep_dist_case	0,431892207809483	0,020818187867986
verbs_tense_dist_Fut	0,427281327000147	0,00229082235621
verbs_num_pers_dist_Sing+3	0,420536227546506	0,005784030048021
upos_dist_INTJ	0,41191670003748	0,000823170132358
verbs_num_pers_dist_Sing+1	0,410580141534613	0,002310891079288
verbs_mood_dist_Sub	0,410515075669655	0,001476742058223
aux_form_dist_Inf	0,405558795453254	0,000914700215332
verbs_form_dist_Part	0,388170371488314	0,006811498033295
aux_mood_dist_Cnd	0,386892144763631	0,000960217415908
upos_dist_ADV	0,369902501897656	0,019581171130525
verbs_form_dist_Ger	0,364639803157933	0,001568539384439
dep_dist_aux	0,360960713317052	0,009055843405076
dep_dist_det:predet	0,359019263933087	0,001951361995464
dep_dist_flat:name	0,357467422521539	0,00524248212462

feature	weightSvm	weightRf
subj_pre	0,343867748737921	0,00562051521167
aux_mood_dist_Sub	0,340877844151919	0,00096100955172
upos_dist_ADP	0,330003946548601	0,020339519532647
dep_dist_nummod	0,325330814151767	0,002233640961262
verbs_num_pers_dist_Plur+1	0,323759975137825	0,003610089773962
verbs_mood_dist_Cnd	0,323574457310054	0
verbs_tense_dist_Pres	0,309855410311883	0,006631224870016
avg_max_depth	0,292374472498807	0,013284002941374
upos_dist_NOUN	0,279490915171777	0,020698348369037
aux_num_pers_dist_Plur+3	0,278083145500673	0,003086215783043
upos_dist_PROPN	0,275371481183988	0,012568387886863
upos_dist_SCONJ	0,261697940335424	0,008802205329619
upos_dist_X	0,25170723263695	0,000167938691594
verbs_mood_dist_Imp	0,249277595144313	0,000859189695712
char_per_tok	0,245266073151555	0,031525185961136
avg_links_len	0,242116305376811	0,029901567710471
upos_dist_DET	0,24057592536667	0,024544831286879
subordinate_post	0,233664680281805	0,004499900366953
dep_dist_root	0,225766113317718	0,028840628045974
dep_dist_obl:agent	0,216348135678266	0,001983254214344
avg_subordinate_chain_len	0,216297553933401	0,005286056127301
subordinate_dist_2	0,215563127856619	0,002358045013555
aux_tense_dist_Fut	0,193711176072946	0,000178480317407
dep_dist_acl:relcl	0,190053856068879	0,010080080159289
dep_dist_expl:pass	0,189735546700015	0,000683086328273
dep_dist_appos	0,185225143668271	0,002223410499384
subordinate_proposition_dist	0,175350106115673	0,007706722907758
aux_tense_dist_Past	0,168951872776873	0,001854936618937
verbs_num_pers_dist_Plur+3	0,152923605707933	0,005087148027286
obj_pre	0,152046574204334	0,003810308982895
subordinate_dist_1	0,145296127636868	0,004655938422155
dep_dist_expl	0,142301446374255	0,009938013417166
obj_post	0,130213330799052	0,004760459085518
verbs_form_dist_Fin	0,12668435407187	0,006617785621224
dep_dist_obl	0,124012027626932	0,018743561818755
verbs_num_pers_dist_Plur+2	0,118700159228041	0,002515630163757
verb_edges_dist_5	0,107710648424096	0,006150334043921
prep_dist_1	0,105076291067718	0,002736854214266
aux_tense_dist_Pres	0,088965640580682	0,005105786615789
aux_num_pers_dist_Sing+3	0,087534657582637	0,004047540399374
upos_dist_CCONJ	0,072409537502772	0,011386357960313
dep_dist_obj	0,069622774447368	0,012019378648736
dep_dist_det:poss	0,067826588630611	0,005775609045142
aux_num_pers_dist_Plur+2	0,06233941515589	0,001918003405043
dep_dist_parataxis	0,058810842248462	0,002131825469786

feature	weightSvm	weightRf
verbs_num_pers_dist_Sing+2	0,058612153134362	0,001090634483898
dep_dist_det	0,056832145938842	0,023519659858858
verb_edges_dist_0	0,056111982576786	0,001848082602317
dep_dist_xcomp	0,052619231502028	0,006695793962321
aux_form_dist_Fin	0,04615487554014	0,003380754093334
avg_max_links_len	0,040760829099156	0,023096334266842
max_links_len	0,040760829099156	0,022372815151101
dep_dist_cc	0,038701573608343	0,010599742498101
dep_dist_advcl	0,028042245594818	0,008261008823893
verbs_tense_dist_Imp	0,025377897804653	0,004986071835358
verbs_form_dist_Inf	0,015460115447411	0,006099875677985
upos_dist_ADJ	0,013605284548973	0,017462149732191
dep_dist_flat:foreign	0,011623554223249	0,000117765680329
dep_dist_acl	0,009358691816347	0,003827274519237
aux_mood_dist_Ind	0,007924743187232	0,003841812596135
aux_tense_dist_Imp	0,006457346002506	0,003613312145325
verbal_head_per_sent	0,00527882207183	0,011658385110968

A.2.2 Dataset *acoust-feats*

Pesi prodotti a seguito dell'addestramento con una *11-fold-cross-validation* di SVM e RandomForest sul dataset *acoust-feats* (contenente solo feature acustiche). I pesi sono stati riordinati in ordine decrescente secondo i valori assegnati da SVM.

feature	weightSvm	weightRf
mfcc_sma[3]_std	1,9897075922359	0,006896615594916
jitterDDP_sma_std	1,7297944269053	0,006185805795811
logHNR_sma_median	1,48695450557256	0,004340692936486
mfcc_sma[5]_std	1,35720893767694	0,006142037610942
pcm_fftMag_fband250-650_sma_mean	1,35426472616079	0,004897103277631
audSpec_Rfilt_sma[18]_std	1,34168009548075	0,004813640082246
audSpec_Rfilt_sma[17]_std	1,31511086273592	0,00539970992321
voicingFinalUnclipped_sma-_median	1,30440581379224	0,005145424528974
pcm_fftMag_spectralEntropy-_sma_std	1,20721687583699	0,005718377751373
mfcc_sma[13]_median	1,20004743202378	0,004884967665617
mfcc_sma[13]_std	1,18890160356426	0,009986873298622
mfcc_sma[8]_std	1,16592602063503	0,009250884380575
mfcc_sma[11]_median	1,16229526507286	0,004562076164293
logHNR_sma_std	1,14851655775732	0,008519105898999
mfcc_sma[3]_mean	1,12805035949663	0,005302027184335
mfcc_sma[3]_median	1,09112988815363	0,004477233687346

feature	weightSvm	weightRf
pcm_zcr_sma_median	1,07719133644946	0,00365862232513
pcm_fftMag_spectralRollOff-75.0_sma_median	1,04363042851222	0,004336766380079
audspec_lengthL1norm_sma-_mean	0,999239999197812	0,002739468003906
audSpec_Rfilt_sma[7]_std	0,992872007338519	0,004839464488907
jitterDDP_sma_median	0,984636435609609	0,004297112788946
mfcc_sma[9]_median	0,973308001595171	0,008206911711914
pcm_fftMag_spectralRollOff-25.0_sma_mean	0,971997767949716	0,004166369804732
audSpec_Rfilt_sma[10]_mean	0,969304595926417	0,004147323383992
mfcc_sma[9]_mean	0,966592126361206	0,007942378104063
voicingFinalUnclipped_sma-_std	0,928135307209459	0,005205091056019
mfcc_sma[11]_mean	0,917809712883013	0,005472092472635
mfcc_sma[10]_mean	0,878629425053475	0,006727029663189
audSpec_Rfilt_sma[4]_std	0,845279500145622	0,005189913176602
mfcc_sma[6]_std	0,837958982167265	0,00717358179733
mfcc_sma[1]_median	0,829976597309724	0,004063024898916
logHNR_sma_mean	0,818579969999917	0,004943971404989
jitterLocal_sma_mean	0,798849684718505	0,005586673291095
audSpec_Rfilt_sma[11]_std	0,789632083136482	0,005904181799326
audSpec_Rfilt_sma[24]_std	0,78357309565834	0,003796556130316
pcm_fftMag_spectralRollOff-90.0_sma_std	0,774017640844193	0,005141522427935
pcm_fftMag_fband1000-4000_sma_std	0,75693321955508	0,003772953361133
pcm_fftMag_spectralHarmoni-city_sma_mean	0,747916386074813	0,004401499592227
mfcc_sma[11]_std	0,74543472947704	0,00504830266127
audSpec_Rfilt_sma[22]-_median	0,742967683412644	0,005375795430877
F0final_sma_mean	0,731103048576301	0,004633852695354
F0final_sma_std	0,728857276986844	0,005810136196686
mfcc_sma[6]_median	0,718875493040457	0,005672146304566
pcm_fftMag_fband250-650_sma_median	0,712230182297773	0,005333538641164
mfcc_sma[2]_median	0,696745200253645	0,005106500944419
pcm_fftMag_spectralSlope-_sma_std	0,66789957711984	0,002723934912864
jitterLocal_sma_median	0,666952351732206	0,003953742937725
F0final_sma_median	0,649457680253533	0,005070950194299
mfcc_sma[2]_mean	0,631395613015172	0,003967117715273
shimmerLocal_sma_std	0,6231475021726	0,004009972430321

feature	weightSvm	weightRf
pcm_fftMag_fband250-650_sma_std	0,622352726693649	0,004643049703328
pcm_fftMag_spectralRollOff-75.0_sma_std	0,598882286991739	0,00392131180991
pcm_fftMag_spectralRollOff-50.0_sma_std	0,597974007058838	0,003976232472713
pcm_fftMag_spectralCentroid-_sma_median	0,5973658142792	0,003518170928065
pcm_fftMag_spectralEntropy-_sma_mean	0,582661364864521	0,003692348049147
audSpec_Rfilt_sma[7]_mean	0,57920786700673	0,007411911535549
audSpec_Rfilt_sma[8]_mean	0,578087102635962	0,004134001310419
audSpec_Rfilt_sma[9]_mean	0,577211646576231	0,005742241809582
mfcc_sma[4]_mean	0,576709172494248	0,003461082724802
audSpec_Rfilt_sma[15]_std	0,573257575107988	0,00642093877482
audSpec_Rfilt_sma[18]_mean	0,569204833915862	0,005047638010395
pcm_RMSenergy_sma_mean	0,558427593866767	0,004247235168803
audSpec_Rfilt_sma[3]_mean	0,544096959070686	0,004894929853217
audSpec_Rfilt_sma[23]-_median	0,531579084719041	0,004924126723645
pcm_zcr_sma_std	0,528394052852775	0,004657451427754
audSpec_Rfilt_sma[19]_mean	0,527861486756095	0,003923710829568
audSpec_Rfilt_sma[25]-_median	0,517245193213483	0,004407536788054
pcm_fftMag_spectralKurtosis-_sma_std	0,514691239079184	0,00558767970765
mfcc_sma[7]_std	0,512158603127773	0,007560121537293
mfcc_sma[13]_mean	0,50909369284571	0,004118784534765
pcm_RMSenergy_sma-_median	0,486873932698785	0,004353807278429
mfcc_sma[6]_mean	0,480395205847259	0,005238089698853
audSpec_Rfilt_sma[9]_std	0,474190207314976	0,008182157563712
mfcc_sma[1]_std	0,469579646168995	0,005481314094208
jitterDDP_sma_mean	0,467908419385978	0,00398565730918
pcm_fftMag_spectralSkewness-_sma_mean	0,465555764113219	0,004141134000575
audSpec_Rfilt_sma[0]_mean	0,463105036232392	0,005419949945994
pcm_fftMag_spectralHarmonicity_sma_std	0,452914788409871	0,003491402552009
mfcc_sma[8]_mean	0,451705186283675	0,00726311996182
pcm_fftMag_spectralSkewness-_sma_std	0,448016131457919	0,003943024340907
mfcc_sma[5]_mean	0,446157388466702	0,004342951751477
mfcc_sma[10]_std	0,446020694292145	0,007556538814447
audSpec_Rfilt_sma[1]_mean	0,445585189262459	0,004881043118872

feature	weightSvm	weightRf
audSpec_Rfilt_sma[20]-_median	0,444664821729788	0,004905898479153
audSpec_Rfilt_sma[24]-_median	0,437583621914371	0,005035559875662
audSpec_Rfilt_sma[14]_std	0,430960251601157	0,004988947201948
audSpec_Rfilt_sma[1]_std	0,42962434793813	0,005333140800076
mfcc_sma[14]_std	0,429367247025397	0,012597083645204
audspec_lengthL1norm_sma-_median	0,426921050940848	0,004245499701877
mfcc_sma[7]_median	0,424859847049618	0,005349960784574
pcm_zcr_sma_mean	0,421543653715673	0,003082446298728
pcm_fftMag_fband1000-4000_sma_mean	0,415564606116355	0,004067558601595
mfcc_sma[8]_median	0,404537572705578	0,008468431941737
shimmerLocal_sma_median	0,398641599327362	0,004056205611039
pcm_fftMag_spectralHarmonicity_sma_median	0,398298599362191	0,005522801468448
pcm_fftMag_psySharpness-_sma_std	0,397301079023634	0,006372057124794
audSpec_Rfilt_sma[11]_mean	0,391484259186399	0,004512252280153
audSpec_Rfilt_sma[2]_std	0,388472620630125	0,006535274272593
mfcc_sma[4]_median	0,386286445131134	0,005847860777709
audSpec_Rfilt_sma[23]_std	0,383686312853357	0,004945143174954
mfcc_sma[4]_std	0,379048771422106	0,006961076886205
pcm_fftMag_spectralRollOff-25.0_sma_std	0,369954200178569	0,003700058603906
audspecRasta_lengthL1norm-_sma_median	0,364502098155654	0,006661517644541
pcm_fftMag_spectralRollOff-25.0_sma_median	0,363501481656513	0,003999129609612
audSpec_Rfilt_sma[18]-_median	0,361713428269233	0,00701923535807
mfcc_sma[14]_mean	0,360775568583648	0,004151852779083
pcm_fftMag_psySharpness-_sma_mean	0,355551041831603	0,002685689551159
pcm_fftMag_spectralFlux-_sma_std	0,340795865627683	0,003999616513626
audSpec_Rfilt_sma[5]_mean	0,33859272862631	0,00558095233164
mfcc_sma[12]_mean	0,335416357146215	0,004493248898192
pcm_fftMag_spectralVariance-_sma_std	0,325697867483001	0,004189332568691
audSpec_Rfilt_sma[6]_mean	0,323454329308561	0,007101029990496
audspecRasta_lengthL1norm-_sma_mean	0,322748552479524	0,007397730030609
audSpec_Rfilt_sma[24]_mean	0,317286198486656	0,005255754876553

feature	weightSvm	weightRf
audSpec_Rfilt_sma[21]_mean	0,31337429344029	0,0041525026466662
audSpec_Rfilt_sma[1]_median	0,307753392742129	0,00586883912682
audSpec_Rfilt_sma[20]_std	0,299678688403588	0,005258514950301
pcm_fftMag_spectralKurtosis-_sma_mean	0,292403796360279	0,00370297782797
pcm_fftMag_spectralSkewness-_sma_median	0,291088344532014	0,004160371947921
audSpec_Rfilt_sma[4]_mean	0,285334597620647	0,004629531526345
pcm_fftMag_spectralSlope-_sma_median	0,277743359577528	0,004273525746036
pcm_fftMag_spectralRollOff-50.0_sma_median	0,27705099154258	0,003673970523807
audSpec_Rfilt_sma[10]-_median	0,259586156787094	0,00428300660105
audSpec_Rfilt_sma[2]_median	0,256951373669501	0,006726127646119
mfcc_sma[12]_std	0,249972471558266	0,007678496544122
audSpec_Rfilt_sma[16]_mean	0,235525874909612	0,004779124154136
audSpec_Rfilt_sma[2]_mean	0,23246245188372	0,003487294988716
pcm_fftMag_spectralRollOff-90.0_sma_median	0,223845063881527	0,003939997449445
shimmerLocal_sma_mean	0,220718372958714	0,007663427731689
audSpec_Rfilt_sma[0]_std	0,218044198649139	0,006678272459285
audSpec_Rfilt_sma[14]-_median	0,209028520571309	0,003859513764392
audSpec_Rfilt_sma[13]_std	0,208678336028527	0,006173499392867
audSpec_Rfilt_sma[13]-_median	0,205688711448971	0,004935139827707
pcm_fftMag_spectralCentroid-_sma_mean	0,203030418642953	0,003596464557096
audSpec_Rfilt_sma[20]_mean	0,201066229601551	0,005311543555056
audSpec_Rfilt_sma[5]_std	0,195555798237137	0,005107707272515
pcm_fftMag_spectralRollOff-75.0_sma_mean	0,194624108112663	0,003547934272344
mfcc_sma[10]_median	0,192852859685559	0,006272939409026
audSpec_Rfilt_sma[8]_median	0,191825156895018	0,006497310627627
pcm_fftMag_spectralCentroid-_sma_std	0,191501329712082	0,004247753662694
audSpec_Rfilt_sma[19]_std	0,187439088874427	0,004030773180607
audSpec_Rfilt_sma[6]_std	0,18657714435507	0,005108830325778
audSpec_Rfilt_sma[4]_median	0,181175502358293	0,006455370359095
pcm_fftMag_psySharpness-_sma_median	0,178244825849163	0,00345640930411
pcm_fftMag_spectralEntropy-_sma_median	0,176248856500933	0,003231275855853

feature	weightSvm	weightRf
pcm_fftMag_spectralRollOff-90.0_sma_mean	0,16519705097727	0,003794219234999
audSpec_Rfilt_sma[17]_mean	0,164622017852722	0,004008668731749
mfcc_sma[1]_mean	0,159757402982848	0,00376007086797
audSpec_Rfilt_sma[25]_std	0,159124652768256	0,003655698936504
pcm_fftMag_spectralRollOff-50.0_sma_mean	0,154017437538414	0,004089332338137
audSpec_Rfilt_sma[22]_mean	0,147843312432713	0,005077785694613
audSpec_Rfilt_sma[12]_mean	0,146028925901874	0,005493392381946
audspecRasta_lengthL1norm_sma_std	0,145536401045106	0,004042307686644
audSpec_Rfilt_sma[8]_std	0,137875199285801	0,005850658260948
audspec_lengthL1norm_sma_std	0,137454351602088	0,004409970365152
audSpec_Rfilt_sma[11]-_median	0,136424685975889	0,004951613071428
audSpec_Rfilt_sma[21]_std	0,124310716946027	0,004098940653098
pcm_fftMag_spectralKurtosis_sma_median	0,124268299654013	0,0043275786362
audSpec_Rfilt_sma[10]_std	0,121725939159802	0,005647286274936
mfcc_sma[12]_median	0,117920065470003	0,003729728281536
audSpec_Rfilt_sma[12]_std	0,107925864216995	0,006162347087588
audSpec_Rfilt_sma[16]_std	0,107647839315675	0,006084160852551
audSpec_Rfilt_sma[3]_std	0,10285074291555	0,005771150494431
pcm_fftMag_spectralFlux_sma_median	0,099598861021107	0,004637529546768
audSpec_Rfilt_sma[22]_std	0,097831524264635	0,005066075250968
audSpec_Rfilt_sma[16]-_median	0,097566262354604	0,008670659550355
mfcc_sma[2]_std	0,084588723453919	0,004133070283706
audSpec_Rfilt_sma[13]_mean	0,083166167766706	0,005197968515503
audSpec_Rfilt_sma[7]_median	0,080818066933851	0,00757355796237
audSpec_Rfilt_sma[23]_mean	0,079070471118442	0,004685931985269
audSpec_Rfilt_sma[9]_median	0,077536071192427	0,006746914196356
voicingFinalUnclipped_sma_mean	0,076468664865615	0,00547708108744
audSpec_Rfilt_sma[0]_median	0,074444400879202	0,005605525381069
audSpec_Rfilt_sma[19]-_median	0,069701512499691	0,004375820093164
audSpec_Rfilt_sma[15]-_median	0,062966545521945	0,004096416261195
audSpec_Rfilt_sma[25]_mean	0,060758531454077	0,004886288041378
audSpec_Rfilt_sma[3]_median	0,056831092639392	0,005415657888097
mfcc_sma[9]_std	0,055569911810409	0,008440051297389

feature	weightSvm	weightRf
pcm_fftMag_spectralVariance-_sma_mean	0,054985462039753	0,003907120819056
pcm_fftMag_spectralFlux-_sma_mean	0,049682364514368	0,004824856437684
pcm_RMSenergy_sma_std	0,04835777390673	0,004228648102166
pcm_fftMag_spectralVariance-_sma_median	0,043551661213329	0,004406199527895
pcm_fftMag_spectralSlope-_sma_mean	0,039798275597377	0,004825225427209
audSpec_Rfilt_sma[15]_mean	0,038938116044662	0,004290897895089
pcm_fftMag_fband1000-4000_sma_median	0,036689948244122	0,004434782730312
mfcc_sma[7]_mean	0,035035980943576	0,005009187302712
audSpec_Rfilt_sma[21]-_median	0,031064998473838	0,003667067502232
mfcc_sma[14]_median	0,028349296550516	0,002938557239112
jitterLocal_sma_std	0,028229541682833	0,004889291374523
audSpec_Rfilt_sma[5]_median	0,028005825958147	0,004314850868862
audSpec_Rfilt_sma[14]_mean	0,027570191176182	0,004389437420473
audSpec_Rfilt_sma[17]-_median	0,021307873748668	0,004567813105249
mfcc_sma[5]_median	0,02026052911873	0,005765619035781
audSpec_Rfilt_sma[12]-_median	0,015699948126169	0,004341311191708
audSpec_Rfilt_sma[6]_median	0,00975419563396	0,00787311286912

A.2.3 Dataset *top-40*

Pesi prodotti a seguito del riaddestramento con una *11-fold-cross-validation* di SVM e RandomForest sul dataset *top-40* (contenente 20 feature linguistiche e 20 feature acustiche selezionate secondo le modalità descritte in 6.5.1). I pesi sono stati riordinati in ordine decrescente secondo i valori assegnati da SVM.

feature	weightSvm	weightRf
audSpec_Rfilt_sma[17]_std	1,9714242756482	0,036402351038534
avg_token_per_clause	1,64462334677052	0,035025726286629
logHNR_sma_std	1,63352834962404	0,046752074989203
pcm_fftMag_fband250-650_sma_mean	1,42311225818195	0,035426714461883
prep_dist_4	1,42206743632157	5,97003316905634E-05
tokens_per_sent	1,40221337761147	0,058032002604227
verbal_root_perc	1,25639277119939	0,003455750331074
dep_dist_csubj	1,12028525150679	0,001935144527875
principal_proposition_dist	1,09346961908366	0,013677955031293
dep_dist_conj	1,09277144203957	0,020679255493024

feature	weightSvm	weightRf
aux_form_dist_Ger	1,06650869402881	0,000553916374176
mfcc_sma[11]_median	1,05941807143623	0,036133692451606
audSpec_Rfilt_sma[18]_std	1,04345774164629	0,030461853003563
dep_dist_amod	1,02661244553934	0,028129469117966
aux_mood_dist_Imp	1	9,21551239370658E-05
subordinate_dist_5	1	0
mfcc_sma[8]_std	0,922772543921127	0,039385001320911
mfcc_sma[5]_std	0,878050542706148	0,037723297521793
mfcc_sma[13]_std	0,839607833224164	0,043781373462395
subordinate_dist_4	0,838703534557139	0,000133431235447
dep_dist_aux:pass	0,680058450867731	0,007673241328549
dep_dist_punct	0,674408417697649	0,033467352070243
pcm_fftMag_spectralEntropy_sma_std	0,670486332407336	0,036017750616112
pcm_fftMag_spectralRollOff75.0_sma_median	0,645359597080159	0,028495502106546
voicingFinalUnclipped_sma_median	0,627800033637811	0,031495941710947
jitterDDP_sma_std	0,621303252697359	0,035142802215463
upos_dist_PUNCT	0,613324673855274	0,034403988137829
audspec_lengthL1norm_sma_mean	0,559898454059805	0,035603229566965
dep_dist_expl:impers	0,514681516924028	0,002398111244147
verb_edges_dist_3	0,502661462035263	0,014035906074319
mfcc_sma[3]_median	0,479310256578771	0,034266488781678
subordinate_dist_3	0,459764323982443	0,000988422373146
mfcc_sma[3]_std	0,392532493354736	0,038187914257019
mfcc_sma[3]_mean	0,33975395847429	0,031046098481522
dep_dist_cop	0,28364752704686	0,013049687136309
pcm_zcr_sma_median	0,118792819445247	0,033606842363363
logHNR_sma_median	0,102538763145958	0,031904370893661
mfcc_sma[13]_median	0,082257684215193	0,032545753169833
avg_prepositional_chain_len	0,063527685652233	0,012100345506217
audSpec_Rfilt_sma[7]_std	0,043218896074805	0,04572938725891

Appendice B

Classificatore di categorie tematiche

B.1 *Classification report*

B.1.1 Scenario 1

Tutte le categorie presenti.

SVM report

	precision	recall	f1-score	support
1	0.21	0.15	0.17	61
2	0.25	0.24	0.24	76
3	0.42	0.44	0.43	157
4	0.51	0.46	0.48	147
5	0.55	0.68	0.60	390
6	0.58	0.49	0.53	92
7	0.51	0.41	0.46	70
8	0.39	0.25	0.31	107
accuracy			0.48	1100
macro avg	0.43	0.39	0.40	1100
weighted avg	0.47	0.48	0.47	1100

RANDOM FOREST report

	precision	recall	f1-score	support
1	0.00	0.00	0.00	61
2	0.00	0.00	0.00	76
3	0.35	0.24	0.29	157
4	0.52	0.29	0.38	147
5	0.42	0.88	0.56	390
6	0.68	0.45	0.54	92
7	0.33	0.01	0.03	70
8	0.43	0.09	0.15	107
accuracy			0.43	1100
macro avg	0.34	0.25	0.24	1100
weighted avg	0.39	0.43	0.35	1100

B.1.2 Scenario 2

Tutte le categorie presenti. Unite le categorie ABCD (storia della Certosa, informazioni storiche, informazioni biografiche e informazioni sui certosini).

SVM report

	precision	recall	f1-score	support
1	0.62	0.72	0.67	436
2	0.60	0.63	0.61	395
3	0.59	0.44	0.51	93
4	0.58	0.43	0.49	70
5	0.45	0.21	0.28	106
accuracy			0.60	1100
macro avg	0.57	0.49	0.51	1100
weighted avg	0.59	0.60	0.59	1100

RANDOM FOREST report

	precision	recall	f1-score	support
1	0.50	0.72	0.59	436
2	0.55	0.57	0.56	395
3	0.66	0.33	0.44	93
4	0.00	0.00	0.00	70
5	0.64	0.07	0.12	106
accuracy			0.52	1100
macro avg	0.47	0.34	0.34	1100
weighted avg	0.51	0.52	0.48	1100

B.1.3 Scenario 3

Eliminata la categoria H (miscellanea).

SVM report

	precision	recall	f1-score	support
1	0.20	0.16	0.18	62
2	0.25	0.18	0.21	76
3	0.43	0.43	0.43	157
4	0.52	0.49	0.51	150
5	0.57	0.68	0.62	396
6	0.69	0.49	0.57	94
7	0.56	0.45	0.50	71
accuracy			0.51	1006
macro avg	0.46	0.41	0.43	1006
weighted avg	0.50	0.51	0.50	1006

RANDOM FOREST report

	precision	recall	f1-score	support
1	0.00	0.00	0.00	62
2	0.00	0.00	0.00	76
3	0.45	0.28	0.35	157
4	0.53	0.27	0.36	150
5	0.45	0.88	0.60	396
6	0.71	0.43	0.53	94
7	0.50	0.03	0.05	71
accuracy			0.47	1006
macro avg	0.38	0.27	0.27	1006
weighted avg	0.43	0.47	0.40	1006

B.1.4 Scenario 4

Unite le categorie ABCD (storia della Certosa, informazioni storiche, informazioni biografiche e informazioni sui certosini). Eliminata la categoria H (miscellanea).

SVM report

	precision	recall	f1-score	support
1	0.66	0.74	0.70	445
2	0.64	0.63	0.63	396
3	0.71	0.48	0.57	94
4	0.62	0.46	0.53	71
accuracy			0.65	1006
macro avg	0.66	0.58	0.61	1006
weighted avg	0.65	0.65	0.65	1006

RANDOM FOREST report

	precision	recall	f1-score	support
1	0.58	0.73	0.65	445
2	0.59	0.58	0.59	396
3	0.81	0.41	0.55	94
4	0.50	0.03	0.05	71
accuracy			0.59	1006
macro avg	0.62	0.44	0.46	1006
weighted avg	0.60	0.59	0.57	1006

B.2 *Confusion matrix*

B.2.1 Scenario 1

Tutte le categorie presenti.

Support Vector Machine

	A	B	C	D	E	F	G	H
A [	9	9	11	5	21	2	0	4]
B [	6	18	11	7	28	2	2	2]
C [	5	12	69	6	48	5	5	7]
D [	5	9	15	68	44	2	2	2]
E [	12	16	35	35	264	6	11	11]
F [	1	5	6	3	20	45	2	10]
G [	0	1	5	3	23	3	29	6]
H [	4	2	12	7	36	13	6	27]

Random Forest

	A	B	C	D	E	F	G	H
A [	0	0	6	3	51	0	0	1]
B [	0	0	10	5	59	2	0	0]
C [	0	0	38	2	112	2	0	3]
D [	0	0	10	43	94	0	0	0]
E [	0	0	18	23	342	4	1	2]
F [	0	0	5	0	40	41	1	5]
G [	0	0	11	4	51	1	1	2]
H [	0	0	10	2	75	10	0	10]

B.2.2 Scenario 2

Tutte le categorie presenti. Unite le categorie ABCD (storia della Certosa, informazioni storiche, informazioni biografiche e informazioni sui certosini).

Support Vector Machine

	ABCD	E	F	G	H
ABCD	[315	99	6	7	9]
E	[121	250	8	8	8]
F	[20	22	41	2	8]
G	[18	18	2	30	2]
H	[36	31	12	5	22]

Random Forest

	ABCD	E	F	G	H
ABCD	[315	116	4	1	0]
E	[166	224	5	0	0]
F	[34	24	31	0	4]
G	[47	21	2	0	0]
H	[69	25	5	0	7]

B.2.3 Scenario 3

Eliminata la categoria H (miscellanea).

Support Vector Machine

	A	B	C	D	E	F	G
A	10	9	11	7	24	1	0
B	8	14	16	6	28	2	2
C	8	8	67	7	59	3	5
D	6	7	9	74	50	2	2
E	14	12	36	43	271	10	10
F	2	3	10	2	25	46	6
G	1	2	8	3	22	3	32

Random Forest

	A	B	C	D	E	F	G
A	0	0	5	3	54	0	0
B	0	0	10	5	59	2	0
C	0	0	44	2	106	5	0
D	0	0	5	41	103	0	1
E	0	0	17	21	349	8	1
F	0	0	8	2	44	40	0
G	0	0	9	4	55	1	2

B.2.4 Scenario 4

Unite le categorie ABCD (storia della Certosa, informazioni storiche, informazioni biografiche e informazioni sui certosini). Eliminata la categoria H (miscellanea).

```
Support Vector Machine
     ABCD   E   F   G
ABCD [329 100   8   8]
E    [128 250   8  10]
F    [ 25  22  45   2]
G    [ 16  20   2  33]
```

```
Random Forest
     ABCD   E   F   G
ABCD [327 113   4   1]
E    [163 229   4   0]
F    [ 30  24  39   1]
G    [ 48  20   1   2]
```

B.3 Ranking feature

Nelle seguenti sottosezioni si riportano i ranking dei dataset maggiormente utilizzati per la fase di analisi. I ranking di tutti gli esperimenti sono consultabili in allegato alla presente Tesi.

B.3.1 Scenario 4

Pesi prodotti a seguito dell'addestramento con una *11-fold-cross-validation* di SVM e RandomForest sul dataset dello scenario 4 che prevede l'unione delle categorie ABCD (storia della Certosa, informazioni storiche, informazioni biografiche e informazioni sui certosini) e la rimozione della categoria H (miscellanea). I pesi sono stati riordinati in ordine decrescente secondo i valori assegnati da SVM.

feature	weightSvm	weightRf
upos_dist_NUM	1,84597996694453	0,001000557982265
dep_dist_flat:name	1,71294197547672	0,001015215908479
mfcc_sma[3]_median	1,54704060106499	0,003960419223859

feature	weightSvm	weightRf
mfcc_sma[6]_mean	1,27631549306746	0,005883551303794
dep_dist_det:predet	1,22450233347244	0,000677877962625
verbs_num_pers_dist_Plur+2	1,21639496435915	0,002677009125182
mfcc_sma[6]_median	1,21074985653914	0,006018943941451
jitterLocal_sma_median	1,19652211694019	0,004519343627539
pcm_fftMag_spectralEntropy-_sma_median	1,19133554471316	0,003335488001114
avg_max_depth	1,18006999416545	0,003341740825032
verbal_head_per_sent	1,14003960109153	0,00250153297402
pcm_fftMag_spectralHarmonicity_sma_std	1,13345018450916	0,006798705745934
dep_dist_amod	1,11725910632345	0,001531646297135
mfcc_sma[10]_std	1,09761930105628	0,005098070169367
mfcc_sma[3]_mean	1,09508825458991	0,004769285195479
pcm_fftMag_spectralFlux-_sma_std	1,09095952468376	0,003979575390927
mfcc_sma[1]_median	1,08583039591866	0,005567405259477
avg_token_per_clause	1,05382373126039	0,005830819199222
F0final_sma_std	1,05275095527463	0,005000736923023
dep_dist_expl:impers	1,018613151935	0,000450655325046
mfcc_sma[9]_mean	1,01720326474413	0,004101685113456
audSpec_Rfilt_sma[0]_std	1,01369795489448	0,003442890709841
upos_dist_SCONJ	1,01070836088614	0,002069599265592
mfcc_sma[11]_mean	1,00958457394206	0,004232626234521
mfcc_sma[2]_std	1,00362430889342	0,004185783177782
dep_dist_ccomp	1,0022004366962	0,000740883913831
prep_dist_4	1	0
verbs_form_dist_Part	0,98868110778804	0,001915616077596
aux_mood_dist_Cnd	0,983630291053121	0,000744151749346
char_per_tok	0,972904488987339	0,004799325422112
mfcc_sma[1]_mean	0,95738332250437	0,006343291018594
dep_dist_obl:agent	0,95190657719023	0,000232897637928
upos_dist_PROPN	0,920444354402612	0,003378897463229
audSpec_Rfilt_sma[4]_std	0,903792707626636	0,00392417207753
dep_dist_iobj	0,900607864883625	0,001305499923571
verbs_tense_dist_Past	0,896253237257099	0,001420425898642
upos_dist_NOUN	0,891149391404852	0,004628358277468
dep_dist_csubj	0,884142019132723	0,000147528749666
verbs_tense_dist_Fut	0,85181125133957	0,002258567908035
pcm_fftMag_fband1000-4000_sma_std	0,851563366772652	0,005731677549308
audSpec_Rfilt_sma[3]_std	0,843322985025075	0,004010695052217
pcm_zcr_sma_median	0,839935892053525	0,002615002801446
verbs_mood_dist_Imp	0,838901243424538	0,000315779085364
mfcc_sma[7]_std	0,837876141470048	0,004648873531239

feature	weightSvm	weightRf
dep_dist_acl:relcl	0,82195216539656	0,002089820951183
mfcc_sma[13]_median	0,818843715058151	0,003966766855548
n_prepositional_chains	0,807588185265047	0,002039314439705
aux_form_dist_Inf	0,796663086075791	0,000257568673058
dep_dist_xcomp	0,788861337159583	0,001620961104243
aux_tense_dist_Pres	0,787491313057146	0,00789843076132
dep_dist_case	0,784393565611722	0,00364930034835
subordinate_proposition_dist	0,783227737339502	0,001940470332372
aux_num_pers_dist_Sing+1	0,779288446996286	0,001564180057707
mfcc_sma[13]_mean	0,771133895134156	0,004551198356412
pcm_RMSenergy_sma_std	0,7543955921888	0,003906139224391
upos_dist_PRON	0,750347439586371	0,00426974717092
jitterDDP_sma_median	0,738160337224866	0,002182301256398
shimmerLocal_sma_mean	0,724496800595318	0,003361919904714
dep_dist_cop	0,723972366074609	0,00200594101256
aux_tense_dist_Fut	0,721461594460205	0
pcm_fftMag_fband250-650_sma_std	0,718339869738781	0,004134715265441
audSpec_Rfilt_sma[5]_std	0,711133790323849	0,003047237041915
avg_prepositional_chain_len	0,697331759224923	0,001012030284778
pcm_fftMag_spectralFlux-_sma_mean	0,696400630947835	0,003845287056445
pcm_fftMag_spectralEntropy-_sma_mean	0,686277385842004	0,003000529006658
mfcc_sma[5]_median	0,683135149099456	0,003023165049616
audSpec_Rfilt_sma[1]_mean	0,678423149767504	0,003941471593008
aux_num_pers_dist_Plur+2	0,678353271359978	0,001658426416055
logHNR_sma_std	0,66825241269089	0,003388488184587
dep_dist_expl	0,660755428413715	0,002451714413578
mfcc_sma[1]_std	0,660051995015664	0,004882637640423
verbs_tense_dist_Imp	0,648219067114638	0,003444774129668
audSpec_Rfilt_sma[5]_median	0,634270172458823	0,005026350766666
dep_dist_det:poss	0,630133880604499	0,000782563766657
aux_num_pers_dist_Plur+1	0,627063002615341	0,005580924508331
subj_pre	0,615508045517942	0,006028738823826
mfcc_sma[11]_std	0,612738651508181	0,002847807280802
audSpec_Rfilt_sma[12]_median	0,612672900513523	0,003084429585359
pcm_fftMag_spectralSlope-_sma_std	0,611514095334528	0,00517648552126
pcm_fftMag_spectralRollOff-90.0_sma_std	0,607055243439419	0,004041779574914
audSpec_Rfilt_sma[4]_median	0,60378554807955	0,004377608728909
tokens_per_sent	0,600210362474911	0,007628503919959
mfcc_sma[9]_median	0,598955991465118	0,003598801300352

feature	weightSvm	weightRf
pcm_fftMag_psySharpness_sma_std	0,597871433347677	0,003118330654627
audSpec_Rfilt_sma[18]_median	0,570593032948921	0,004582959034047
avg_max_links_len	0,563126249052729	0,002951234603862
max_links_len	0,563126249052729	0,003537970737428
dep_dist_obl	0,556800676698529	0,002879877634742
aux_mood_dist_Imp	0,554067940971659	0
audSpec_Rfilt_sma[23]_median	0,552679374102858	0,004662572887513
audSpec_Rfilt_sma[0]_median	0,539939785868725	0,003065466083848
mfcc_sma[12]_median	0,536616937734394	0,003575607065923
pcm_fftMag_spectralKurtosis_sma_mean	0,528494057683595	0,003992718800103
pcm_fftMag_spectralRollOff50.0_sma_mean	0,525452755292754	0,002684749665042
audSpec_Rfilt_sma[7]_std	0,508983033741863	0,003660138444106
pcm_fftMag_spectralSlope_sma_median	0,505903031775375	0,002903246782993
aux_num_pers_dist_Sing+2	0,5	0
aux_num_pers_dist_Sing+3	0,49816126404842	0,001811395889021
mfcc_sma[14]_mean	0,496252581449966	0,008025401680313
mfcc_sma[3]_std	0,490654644432709	0,008320665336624
audSpec_Rfilt_sma[19]_mean	0,488209083125163	0,003863798241311
mfcc_sma[6]_std	0,487398701987488	0,003847924867676
audSpec_Rfilt_sma[6]_median	0,470464078195448	0,00435547289155
pcm_fftMag_spectralEntropy_sma_std	0,463629462138101	0,004269349817445
audSpec_Rfilt_sma[19]_std	0,45760541740907	0,004905215306118
audSpec_Rfilt_sma[24]_std	0,457154926939076	0,005907283511313
pcm_fftMag_spectralRollOff75.0_sma_median	0,45490771186094	0,002503315036866
verbs_mood_dist_Sub	0,454720090902192	7,49432084399566E-05
audSpec_Rfilt_sma[18]_std	0,452552789369676	0,004034278368302
aux_form_dist_Part	0,445535928787477	0,000365575534114
mfcc_sma[7]_mean	0,437391651584832	0,002907444625661
pcm_zcr_sma_mean	0,436938110656854	0,002005810282647
mfcc_sma[9]_std	0,436064785597495	0,003919072936693
verbs_mood_dist_Ind	0,431462448884645	0,000969276684132
mfcc_sma[14]_std	0,430634727766744	0,005250274331352
subordinate_dist_3	0,429478529163309	0,000112615824558
audSpec_Rfilt_sma[9]_std	0,42906956272104	0,003083828102982
jitterDDP_sma_mean	0,428867934517207	0,003269856208039
audSpec_Rfilt_sma[9]_mean	0,42454508970733	0,003313195602478
subj_post	0,418131298265664	0,004785272911724
pcm_fftMag_spectralRollOff50.0_sma_median	0,416188000754232	0,002830176059532

feature	weightSvm	weightRf
audSpec_Rfilt_sma[24]_mean	0,415888942940789	0,004006688222098
mfcc_sma[8]_mean	0,415754735206491	0,004856643840441
mfcc_sma[5]_std	0,413027523081269	0,004385116736082
mfcc_sma[5]_mean	0,411722807755012	0,004504705375585
aux_tense_dist_Imp	0,410106078822395	0,004723688156851
audSpec_Rfilt_sma[2]_mean	0,409122072389891	0,004531142121168
audSpec_Rfilt_sma[25]_std	0,408548209238973	0,004244067445026
verbs_num_pers_dist_Sing+1	0,405105917972506	0,001732929781666
audSpec_Rfilt_sma[1]_std	0,403124258058767	0,004425125318131
audSpec_Rfilt_sma[22]_std	0,399614880011725	0,004008811149347
pcm_fftMag_spectralRollOff-75.0_sma_mean	0,398767589682976	0,002746390459482
pcm_fftMag_spectralKurtosis-_sma_std	0,395861174219	0,004923420806975
voicingFinalUnclipped_sma-_median	0,393296951217565	0,003640186918982
audSpec_Rfilt_sma[16]_median	0,386483412084914	0,003438546722484
upos_dist_AUX	0,386444024465817	0,001900271897085
F0final_sma_mean	0,385575422179159	0,003714297866813
aux_form_dist_Fin	0,38430113478006	0,001638620668977
avg_subordinate_chain_len	0,382348649571554	0,001501277078329
aux_form_dist_Ger	0,37850647548815	0,000225958588237
mfcc_sma[8]_median	0,377048080832338	0,003849203001669
audSpec_Rfilt_sma[15]_std	0,36882078250872	0,003887081104929
verbs_num_pers_dist_Plur+1	0,364304903896432	0,004172622771045
dep_dist_mark	0,360880727034491	0,001874166618178
upos_dist_ADP	0,359387920194436	0,004053656535977
jitterDDP_sma_std	0,35655820189973	0,005030363175572
mfcc_sma[4]_median	0,353100115720878	0,002632689546436
audSpec_Rfilt_sma[24]_median	0,352026793137881	0,003182743676937
pcm_fftMag_spectralFlux_sma-_median	0,347720441756493	0,00381948016057
verbs_num_pers_dist_Sing+3	0,342175816938692	0,002245520189247
dep_dist_nsubj:pass	0,341538453926631	0,000355669702493
aux_num_pers_dist_Plur+3	0,341202332210777	0,000815033437405
dep_dist_nmod	0,337334703452861	0,001798911617988
verb_edges_dist_6	0,336470254337627	0,000570741353412
audSpec_Rfilt_sma[8]_std	0,336357948945853	0,003964032122955
prep_dist_3	0,333366981642118	2,72900410323849E-05
mfcc_sma[11]_median	0,332670783409071	0,003636287790256
pcm_fftMag_spectralHarmoni-city_sma_mean	0,330893845708168	0,00392509514521
shimmerLocal_sma_std	0,330667032013835	0,004689663045837
pcm_fftMag_spectralSkewness-_sma_std	0,328285600895839	0,004417849893266

feature	weightSvm	weightRf
subordinate_post	0,325524190053244	0,000380779054225
audSpec_Rfilt_sma[3]_mean	0,323806373944926	0,00512319759739
audspecRasta_lengthL1norm-_sma_std	0,322166450358118	0,003198973346908
pcm_fftMag_spectralVariance-_sma_std	0,319574623741772	0,004350049158158
audSpec_Rfilt_sma[16]_mean	0,312462705666348	0,002581180959884
mfcc_sma[12]_std	0,301435233240326	0,00586967712399
dep_dist_conj	0,299201098511126	0,00148285982494
logHNR_sma_median	0,296067777417477	0,002504800164311
dep_dist_advmod	0,293070663634651	0,003508861370588
voicingFinalUnclipped_sma-_mean	0,291598701578721	0,002841869812768
audSpec_Rfilt_sma[1]_median	0,291526534226982	0,003047688749982
audSpec_Rfilt_sma[8]_median	0,291350275970442	0,004682873406991
dep_dist_cc	0,29006980141769	0,001303051899618
audSpec_Rfilt_sma[15]_median	0,289772878618749	0,006852610815541
verb_edges_dist_2	0,287573816599412	0,001969934008979
dep_dist_advcl	0,279209920176811	0,000923099648848
pcm_fftMag_spectralCentroid-_sma_mean	0,278484158052976	0,002666253311692
pcm_fftMag_spectralCentroid-_sma_median	0,277866525057867	0,002998794837685
pcm_RMSenergy_sma_median	0,277844561213932	0,00332913618778
dep_dist_nummod	0,275179571804979	0,000417936955133
upos_dist_X	0,27501909854851	0,000114231926172
audSpec_Rfilt_sma[14]_median	0,272449534478874	0,004078931599763
verbs_form_dist_Ger	0,271528694125174	0,00015859768144
pcm_fftMag_psySharpness-_sma_median	0,271392660072877	0,002640452944475
pcm_fftMag_spectralKurtosis-_sma_median	0,270401908035442	0,003789513292502
audSpec_Rfilt_sma[22]_median	0,270350386118423	0,005506125269875
verbs_form_dist_Inf	0,270050152318191	0,001911489431147
upos_dist_DET	0,266540933194619	0,003933225530174
audSpec_Rfilt_sma[21]_median	0,263010490550499	0,004666012527531
audSpec_Rfilt_sma[23]_mean	0,2624466769294	0,004341846835323
pcm_fftMag_spectralRollOff-25.0_sma_median	0,259530951485292	0,002163118779817
jitterLocal_sma_mean	0,259182085655752	0,003564509481964
verb_edges_dist_3	0,258674903269863	0,001205052133832
subordinate_dist_2	0,249631771817707	0,00074153567392
upos_dist_VERB	0,247843738075723	0,003764421428884
audSpec_Rfilt_sma[13]_std	0,245199256163119	0,003573125620623
dep_dist_compound	0,241976145296828	0,000274089690725

feature	weightSvm	weightRf
audSpec_Rfilt_sma[17]_mean	0,239329753240572	0,003604390491251
pcm_RMSenergy_sma_mean	0,237315933437571	0,004928269220965
mfcc_sma[7]_median	0,232042488890471	0,005672844835414
audSpec_Rfilt_sma[18]_mean	0,22959563115424	0,004243807031252
audSpec_Rfilt_sma[21]_std	0,22772960767648	0,002938318265864
audSpec_Rfilt_sma[15]_mean	0,227512342655157	0,003455455624364
obj_post	0,226546978552719	0,00172474353929
dep_dist_acl	0,224236206055435	0,00060799066238
verbs_num_pers_dist_Plur+3	0,223019280813709	0,000718102505765
audSpec_Rfilt_sma[25]_mean	0,219574494844728	0,004453033582223
audSpec_Rfilt_sma[11]_median	0,218069531591375	0,003821414527524
pcm_fftMag_spectralSlope-_sma_mean	0,216769334392893	0,004591768962614
jitterLocal_sma_std	0,216212397453859	0,003081349021704
pcm_fftMag_psySharpness-_sma_mean	0,214951667298152	0,003673730538619
audSpec_Rfilt_sma[19]_median	0,205505394190503	0,003338792053251
verbs_num_pers_dist_Sing+2	0,20524474887635	0,000178027349994
audSpec_Rfilt_sma[10]_mean	0,204659484345157	0,003230509053262
dep_dist_appos	0,203105349556406	0,000485777068276
audSpec_Rfilt_sma[12]_std	0,197148845440765	0,003227332235669
audSpec_Rfilt_sma[5]_mean	0,195868133760115	0,002448041995673
pcm_fftMag_spectralCentroid-_sma_std	0,194392150903283	0,002828076238164
dep_dist_aux	0,193889209534188	0,00235005678464
principal_proposition_dist	0,193044476410236	0,0016743903943
dep_dist_fixed	0,190952457410119	0,000662781439924
mfcc_sma[10]_mean	0,189179745042225	0,003423790473777
verb_edges_dist_4	0,189127835625726	0,00078580900281
obj_pre	0,188714063084539	0,0008703140539
dep_dist_obj	0,187595741481339	0,002844243595246
pcm_fftMag_spectralSkewness-_sma_median	0,183662025214716	0,00243406981682
lexical_density	0,182396075038923	0,003135657392216
verbs_form_dist_Fin	0,181181594684318	0,00130768590279
audSpec_Rfilt_sma[11]_mean	0,180913818411	0,003118433345106
upos_dist_CCONJ	0,17895869030658	0,00215906475345
pcm_fftMag_fband250-650_sma_mean	0,177522112582892	0,003531943251051
pcm_fftMag_spectralRollOff-50.0_sma_std	0,177064749903792	0,003932193017446
logHNR_sma_mean	0,176954986154044	0,003238238027075
pcm_zcr_sma_std	0,174610413646832	0,00353839575057
verb_edges_dist_1	0,173811129634892	0,000783944747552

feature	weightSvm	weightRf
audspecRasta_lengthL1norm-_sma_mean	0,171178114607812	0,003491016853134
subordinate_dist_4	0,169797856482479	2,99738730057696E-05
audSpec_Rfilt_sma[21]_mean	0,16930906895734	0,004476750829755
mfcc_sma[2]_mean	0,167379455436901	0,002787259009379
audSpec_Rfilt_sma[22]_mean	0,167337169333273	0,003731466510447
pcm_fftMag_spectralSkewness-_sma_mean	0,164778703847503	0,002741744471589
aux_tense_dist_Past	0,164708520292813	0,000479377983202
prep_dist_2	0,164641756047025	0,000659241550795
aux_mood_dist_Ind	0,164461612896702	0,000814426660278
audSpec_Rfilt_sma[20]_std	0,1607636570496	0,003379002696637
audspecRasta_lengthL1norm-_sma_median	0,158189971804617	0,003124073089408
dep_dist_discourse	0,151515151515152	0,000187041775346
audSpec_Rfilt_sma[20]_mean	0,151098772027339	0,003120023303187
pcm_fftMag_spectralRollOff-75.0_sma_std	0,148570552017105	0,003980009743097
audSpec_Rfilt_sma[8]_mean	0,146846867494872	0,003274779393695
audSpec_Rfilt_sma[14]_std	0,142728239600579	0,004287984361647
audSpec_Rfilt_sma[7]_median	0,140420990709341	0,003581403810525
verb_edges_dist_0	0,139911189938097	0,000250339783285
audSpec_Rfilt_sma[17]_median	0,137911070221683	0,003190214802145
dep_dist_parataxis	0,129415605889721	0,000346749345411
mfcc_sma[14]_median	0,128166899438526	0,003824277703955
audSpec_Rfilt_sma[12]_mean	0,125584142329458	0,002712256258331
prep_dist_1	0,118711493920173	0,000949370076674
mfcc_sma[2]_median	0,115391218412441	0,00387480930007
mfcc_sma[10]_median	0,115306403911603	0,003053789547886
audSpec_Rfilt_sma[6]_mean	0,110004967765249	0,002903329094581
mfcc_sma[12]_mean	0,107264310564659	0,004106200347383
dep_dist_flat:foreign	0,107142857142857	2,91835734432443E-05
audSpec_Rfilt_sma[23]_std	0,101693041673165	0,003058390103489
upos_dist_ADJ	0,100571521916184	0,002645811168192
F0final_sma_median	0,099273147493193	0,003400075448803
verb_edges_dist_5	0,09327963689362	0,000701057037668
audSpec_Rfilt_sma[4]_mean	0,086249332584643	0,004353266964617
aux_mood_dist_Sub	0,085239082459919	8,02726708161643E-05
subordinate_pre	0,08515155227667	0,000643293192797
verbal_root_perc	0,083777795464556	0,000591869039074
audSpec_Rfilt_sma[7]_mean	0,081797165089675	0,003137454330635
audSpec_Rfilt_sma[16]_std	0,081364796676134	0,002777843097952
pcm_fftMag_spectralVariance-_sma_median	0,079826705932888	0,003348557882783
dep_dist_root	0,078677969933789	0,006692851838518

feature	weightSvm	weightRf
avg_verb_edges	0,07356839198205	0,002106173416444
pcm_fftMag_fband1000-4000_sma_median	0,073060645128733	0,002638349156137
audSpec_Rfilt_sma[9]_median	0,072598034038521	0,003552059850314
audSpec_Rfilt_sma[2]_std	0,070080249318217	0,00374274298613
audSpec_Rfilt_sma[13]_median	0,069016198168615	0,003869290571813
audSpec_Rfilt_sma[3]_median	0,068643371155591	0,005020024566194
mfcc_sma[4]_std	0,067673240305425	0,005989576242921
mfcc_sma[4]_mean	0,067517639318794	0,003538631596415
subordinate_dist_1	0,06103112850181	0,000894362346133
upos_dist_INTJ	0,060606060606061	0,003434208003604
dep_dist_nsubj	0,059531917621939	0,002568484612439
pcm_fftMag_spectralRollOff-90.0_sma_mean	0,058764633632649	0,003568656053607
dep_dist_expl:pass	0,058571658592389	0,000215842703116
verbs_tense_dist_Pres	0,056970704478147	0,0018042120794
audSpec_Rfilt_sma[10]_std	0,056789424368653	0,004133478271437
pcm_fftMag_spectralHarmonicity_sma_median	0,055055486950368	0,003090566738323
pcm_fftMag_fband250-650_sma_median	0,053489570059709	0,00327057190999
pcm_fftMag_fband1000-4000_sma_mean	0,052766044351188	0,004171843233108
dep_dist_aux:pass	0,049805279258715	0,000761995035029
audSpec_Rfilt_sma[11]_std	0,047504638151459	0,001884603502939
audSpec_Rfilt_sma[6]_std	0,045825810253476	0,003686365280467
mfcc_sma[8]_std	0,043566788203265	0,005984605913603
pcm_fftMag_spectralRollOff-25.0_sma_mean	0,042439442154944	0,002850366493326
audSpec_Rfilt_sma[25]_median	0,040121197684314	0,00342195134984
voicingFinalUnclipped_sma_std	0,040042743858578	0,003514858534809
pcm_fftMag_spectralVariance_sma_mean	0,037139504051282	0,003085077222205
audSpec_Rfilt_sma[13]_mean	0,03604375743458	0,002391863335914
upos_dist_PUNCT	0,029826236566557	0,009023949956985
dep_dist_punct	0,029826236566557	0,007561050526774
audSpec_Rfilt_sma[14]_mean	0,027331584329755	0,003838287705398
engagement	0,026506212697001	0,000519922850115
audspec_lengthL1norm_sma_mean	0,025271578701464	0,003752679165865
mfcc_sma[13]_std	0,023608149438914	0,006749293378157
audspec_lengthL1norm_sma_median	0,021280136637095	0,003424705376732
audSpec_Rfilt_sma[10]_median	0,019773274868893	0,003479909147218
audSpec_Rfilt_sma[2]_median	0,017572829579542	0,004493039437576

feature	weightSvm	weightRf
pcm_fftMag_spectralRollOff-90.0_sma_median	0,017178874418207	0,002681465412061
upos_dist_ADV	0,016274672943553	0,002983920631427
pcm_fftMag_spectralRollOff-25.0_sma_std	0,014541198475868	0,002966710179418
audSpec_Rfilt_sma[0]_mean	0,013506911087056	0,005514402791562
avg_links_len	0,013402396665711	0,005781652369857
dep_dist_det	0,012725212497117	0,00558789670712
audspec_lengthL1norm_sma-_std	0,011816508690842	0,003182442244258
shimmerLocal_sma_median	0,007957932598032	0,002311941921669
audSpec_Rfilt_sma[17]_std	0,005638662765605	0,00387455524238
audSpec_Rfilt_sma[20]_median	0,003559597058832	0,003829221189814
verbs_mood_dist_Cnd	0	0
subordinate_dist_5	0	2,90145358834979E-05

B.3.2 Dataset *top-40-cat*

Pesi prodotti a seguito del riaddestramento con una *11-fold-cross-validation* di SVM e RandomForest sul dataset *top-40-cat* (contenente le migliori 40 feature selezionate secondo le modalità riportate in 7.5.1). I pesi sono stati riordinati in ordine decrescente secondo i valori assegnati da SVM.

feature	weightSvm	weightRf
dep_dist_flat:name	2,73470381931461	0,009001347816967
mfcc_sma[11]_mean	2,7213393704065	0,036676926949414
mfcc_sma[1]_median	2,60023169420342	0,043037583766234
pcm_fftMag_spectralEntropy-_sma_median	2,55760157321487	0,038348183319972
upos_dist_NUM	2,4438702081237	0,005405208362626
mfcc_sma[6]_mean	2,41452485625791	0,040025066690652
pcm_fftMag_spectralHarmoni-city_sma_std	2,37675469196411	0,042351783602517
mfcc_sma[3]_median	1,99513530535458	0,033911186624135
verbs_num_pers_dist-_Plur+2	1,97052895158832	0,011836391362767
upos_dist_PROPN	1,86360100774201	0,0223050775981
F0final_sma_std	1,76752257933714	0,040566062990679
verbal_head_per_sent	1,74603704436611	0,01899815940094
mfcc_sma[1]_mean	1,60581845136642	0,044282032379196
avg_max_depth	1,5980523944916	0,030800905283977
pcm_fftMag_fband1000-4000_sma_std	1,59386603338812	0,038173977919336
mfcc_sma[10]_std	1,47816389379179	0,041356606924357
dep_dist_csubj	1,3040347511111	0,001399806810797

feature	weightSvm	weightRf
prep_dist_4	1,29699067262304	0,000151476862275
mfcc_sma[3]_mean	1,27176309191833	0,034129341791838
verbs_form_dist_Part	1,22110394110344	0,012512000460771
verbs_tense_dist_Past	1,18368077848835	0,010433339796605
dep_dist_iobj	1,17817954057547	0,006554474101216
mfcc_sma[6]_median	1,1628368516719	0,038301126124613
dep_dist_amod	1,15903665107565	0,018975648181902
jitterLocal_sma_median	1,11071711217151	0,028594106315747
dep_dist_obl:agent	1,07618982071692	0,003044080591692
upos_dist_SCONJ	1,01920182882731	0,014843135477705
dep_dist_expl:impers	0,907888641897118	0,001619982551459
aux_mood_dist_Cnd	0,80129571177569	0,001995446489744
dep_dist_ccomp	0,779677746151329	0,006135718874805
dep_dist_det:predet	0,739986657509056	0,003159563216407
mfcc_sma[9]_mean	0,701493608850825	0,033780987589935
pcm_fftMag_spectralFlux-_sma_std	0,66984899041114	0,033793522772605
avg_token_per_clause	0,591740767950299	0,040853343803956
verbs_tense_dist_Fut	0,567246571268199	0,010732646809213
audSpec_Rfilt_sma[0]_std	0,522503296405134	0,037468940916534
mfcc_sma[2]_std	0,410939416687512	0,036852323598434
char_per_tok	0,355388243933788	0,043102237411073
upos_dist_NOUN	0,264525021838921	0,041248404789102
audSpec_Rfilt_sma[4]_std	0,15616189465473	0,043241843669705