



UNIVERSITÀ DEGLI STUDI DI FIRENZE
SCUOLA DI INGEGNERIA - DIPARTIMENTO DI INGEGNERIA
DELL'INFORMAZIONE

Tesi di Laurea Triennale in Ingegneria Informatica

**RETI NEURALI PER L'ANALISI DI SCRITTURA
CON SMART-PEN**

Candidato
Emanuele Vivoli

Relatore
Prof. Simone Marinai

Correlatore
Prof.ssa Tamara Zappaterra

Anno Accademico 2017/2018

La teoria è quando si sa tutto ma non funziona niente. La pratica è quando tutto funziona ma non si sa il perché. In ogni caso si finisce sempre con il coniugare la teoria con la pratica: non funziona niente e non si sa il perché.

- Albert Einstein.

Indice

Ringraziamenti	i
Introduzione	iii
1 Background	1
1.1 Background Disturbi Specifici dell’Apprendimento	1
1.1.1 Disgrafia	2
1.1.2 Disortografia	3
1.1.3 Test per Disgrafia e Disortografia	4
1.2 Reti Neurali Artificiali	10
1.2.1 Reti Neurali artificiali	11
1.2.2 Motivazioni biologiche	11
1.2.3 Architettura di una rete neurale	13
1.2.4 Tipi di reti neurali	14
1.3 Background Analisi della scrittura	17
1.3.1 Scrittura Online ed Offline	18
1.4 Background Tecnologie	19
1.4.1 I Digitalizzatori Passivi	20
1.4.2 I Digitalizzatori Attivi	22

2 Il Progetto	24
2.1 Estrazione dati	25
2.1.1 Formato file WILL	27
2.1.2 Formato dati WILL	27
2.1.3 Decodifica file protobuf	29
2.1.4 Esempio utilizzo	31
2.2 Elaborazione degli stroke	32
2.2.1 Rappresentazione degli stroke	32
2.2.2 Creazione Parole e Righe dagli stroke	33
2.2.3 Riconoscimento del testo	34
2.2.4 Interpretazione del testo BHK	35
2.3 Creazione Features	39
2.3.1 BHK Standard	40
2.3.2 BHK modificato	42
2.3.3 Metodo Offline	46
3 Dataset	48
3.1 Dataset BHK standard	51
3.2 Dataset BHK modificato	51
3.3 Dataset CNN	52
4 Esperimenti e Risultati	55
4.1 Esperimenti BHK standard	55
4.1.1 BHK standard sul dataset degli universitari	56
4.1.2 BHK standard sul dataset dei bambini	57
4.2 Esperimenti BHK modificato	61
4.2.1 BHK modificato sul dataset degli universitari	61
4.2.2 BHK modificato sul dataset dei bambini	63

4.3 Esperimenti CNN	65
4.3.1 CNN sul dataset degli universitari	68
4.3.2 CNN sul dataset dei bambini	70
5 Conclusioni e sviluppi futuri	73
Elenco delle figure	i
Elenco delle tabelle	ii
Bibliografia	iii

Ringraziamenti

Questo è il capitolo dei ringraziamenti: l'ultima cosa da scrivere dopo alle conclusioni e subito prima di stampare il tutto. È un capitolo importante perchè conclude un percorso che ha portato gioie e preoccupazioni, notti a studiare fino alle 2 ma anche mesi a camminare e viaggiare liberi da tutti i pensieri, perchè gli esami erano andati ed andati bene. Nella prima tesi che ho letto di qualcun'altro ho solo sfogliato le figure e letto i ringraziamenti; io di figure ne ho messe tante ed alcune anche colorate, adesso non resta che fare dei bei ringraziamenti.

Innanzitutto ringrazio gli amici di scuola, prima i maschietti: Svitèn, lo Zio, King, il Delù, Salapower, il Bio ed anche il Bazza ormai che ci siamo e poi le femminucce: l'Elis, la Franci e la Miche che mi hanno dato sostegno dall'inizio alla fine e mi hanno fatto passare tre anni (tre anni ?!?!) davvero belli nonostante alcune materie insostenibili. Qualcuno più, qualcuno meno, hanno contribuito alla mia crescita, ad un cambiamento del mio modo di pensare che sicuramente è una cosa grossa. Prima ero più Sciallo, ora sono più Ingegnere; non è per niente una cosa negativa, va solo saputa gestire.

Ringrazio i miei genitori che non mi hanno mai fatto mancare niente, ed i miei fratelli per aver tenuto occupate le lamentele dei miei tutte per loro, che sia per le molteplici bocciature o multe, che sia per il non prendere appunti e rispondere maleducatamente. Ho passato i miei studi come in un albergo,

e non è affatto male.

Ringrazio tutti quelli che mi sono stati accanto, per scelta o per obbligo, a cui necessariamente ho rotto l'anima con le mie cavolate; i professori bravi e qualche assistente; i miei nonni che mi insegnano ogni volta l'arte della pazienza ed anche non sapendolo mi mostrano la potenza della vecchiaia e del cambiamento; la famiglia Graziani che mi ha inglobato in svariate cene e giornate insieme (anche concertoni), li ringrazio per tutto quello che hanno fatto per me.

Ringrazio poi tutti quelli che mi danno ispirazione, gli amici i parenti e soprattutto chi mi da una foga tremenda: grazie Gino continua così che prima o poi si sfonda.

Ringrazio infine altre due persone, perchè tre anni sono lunghi e tre anni fa precise iniziava l'università in parallelo con un altro avvenimento importante. Ad oggi questo avvenimento importante mi sostiene quando inciampo, mi blocca e mi corregge quando sbaglio e brinda con me quando faccio bene, quando raggiungo i miei risultati; un grazie con la 'G' maiuscola:

Grazie Lindi.

Sono molto soddisfatto del percorso fatto fino ad ora e spero di poter crescere ancora molto; non si finisce mai di imparare.

Buona lettura.

Introduzione

Questo progetto si pone l'obbiettivo di creare uno strumento di supporto agli esperti in ambito di DSA per offrire un giudizio imparziale alle loro decisioni ed analizzare dal punto di vista tecnologico la scrittura. Più approfonditamente tratteremo, sia metodi automatici di analisi di scrittura, sia tecnologie di analisi basate su reti neurali artificiali. La struttura di questo elaborato indirizza il lettore a comprendere meglio le attuali tecniche di analisi della scrittura dei bambini DSA e le moderne tecniche di intelligenza artificiale per modellare problemi complessi, come appunto questo della scrittura a mano.

Il Capitolo 1 cercherà di introdurre il concetto di DSA anche mediante alcune tra le definizioni più classiche, i concetti di reti neurali artificiali ed un accenno alle tecnologie moderne di inchiostro digitale; questi concetti risulteranno fondamentali per la comprensione dei capitoli successivi. Si introducirà poi ciò che è stato svolto durante il progetto di tesi nel successivo Capitolo 2, a partire dall'estrazione dei dati codificati dalla penna digitale e la loro elaborazione mediante un modello di test largamente utilizzato dai colleghi di Scienze della Formazione per quanto concerne i disturbi specifici dell'apprendimento. Il test è chiamato BHK e verrà data, sempre nello stesso capitolo, una sua implementazione algoritmica. Con il Capitolo 3 si definiscono i dataset costruiti, cioè un primo dataset composto da documenti scritti con la

mano principale e secondaria da studenti universitari, ed un secondo dataset raccolto con l'aiuto della scuola media del Comprensivo di Scarperia e San Piero distinguendo bambini DSA e non DSA. Nel successivo Capitolo 4 verranno presentati gli esperimenti effettuati su entrambi i dataset, analizzati i risultati ed espresso un parere su questi, positivi o negativi che siano. Si presenta poi l'ultimo capitolo (Capitolo 5) dove tramite delle conclusioni sul lavoro svolto si prova ad introdurre idee per eventuali sviluppi futuri.

Spero che questo mio lavoro sia interessante, fornisca una visione trasversale sul problema dell'analisi della scrittura e che sia un punto di partenza per entrare in una dimensione dove le nuove tecnologie sono impiegate per risolvere i più disparati problemi del mondo.

Capitolo 1

Background

1.1 Background Disturbi Specifici dell’Apprendimento

In psicologia cognitiva con la locuzione disturbi specifici di apprendimento (sigla DSA) si intendono disturbi nell’apprendimento di alcune abilità specifiche che non permettono una completa autosufficienza nell’apprendimento stesso, poiché le difficoltà si sviluppano sulle attività che servono per la trasmissione della cultura come, ad esempio, la lettura (dislessia), la scrittura (disgrafia, disortografia) e/o il far di conto (discalculia). Fanno parte della famiglia dei Disturbi Evolutivi Specifici. I disturbi specifici di apprendimento si verificano in soggetti che hanno intelligenza almeno nella norma, con caratteristiche fisiche e mentali nella norma, ma con un funzionamento scolastico deficitario. [1] [2] [3] [4]

Piuttosto che sulla la dislessia, nota come disturbo della lettura, la nostra attenzione si concentra più su altri fattori che coinvolgono la scrittura, e dunque sulle problematiche di Disgrafia e Disortografia, che trattiamo più

approfonditamente in seguito.

Questi parametri forniscono, ad una figura specializzata, le caratteristiche della scrittura di un bambino sulle quali dare più attenzione, per diagnosticare o meno dei fattori disgrafici e disortografici.

1.1.1 Disgrafia

La disgrafia è una difficoltà specifica nella riproduzione dei segni alfabetici e numerici, il cui tracciato appare incerto, irregolare nella forma e nella dimensione, e comunque non adeguato ai modelli [5]. I criteri per riconoscere una scrittura disgrafica sono basati su alcuni parametri, quali:

- velocità di scrittura;
- impugnatura scorretta della penna;
- pressione (debole o eccessiva) esercitata sul foglio;
- tendenza alla macro o micrografia;
- discontinuità nel gesto ed inversioni (anche da destra verso sinistra);
- ritoccatura del segno già tracciato;
- direzione del gesto grafico;
- andamento del tracciato: scrittura in senso orario anziché antiorario;
- occupazione dello spazio nel foglio;
- inesatta legatura dei segni;
- distanza tra le parole;
- caratteristiche delle produzioni e riproduzioni grafiche;

- caratteristiche dell'esecuzione di copie di frasi o disegni geometrici;

1.1.2 Disortografia

La disortografia è un disturbo specifico della scrittura che non rispetta regole di trasformazione del linguaggio parlato in linguaggio scritto, non imputabile alla mancanza di esperienza o a deficit motori o sensoriali [5].

I sintomi della disortografia possono essere omissioni di grafemi o parti di parola (es. pote per ponte o camica per camicia), sostituzioni di grafemi (es. vaccia per faccia; parde per parte), inversioni di grafemi (es. il per li; spicologia per psicologia).

Questo tipo di difficoltà porta a non tradurre correttamente i suoni che compongono le parole in simboli grafici; gli errori che vengono commessi possono essere così distinti:

- confusione fra fonemi simili (f-v, d-t, b-p, l-r, ecc..)
- confusione fra grafemi simili (b-p), cioè che hanno stessa forma
- omissioni (palla-pala, fuoco-foco, cartolina-carolina), cioè con doppia consonante, vocale intermedia, consonante intermedia
- inversioni nella sequenza di suoni all'interno delle parole (sefamoro anzichè semaforo)

La disortografia può derivare da una difficoltà di linguaggio, da scarse capacità di percezione visiva e uditiva, da un'organizzazione spazio-temporale non ancora sufficientemente acquisita, da un processo lento nella simbolizzazione grafica. È un disturbo che si presenta quando dalla forma si passa al contenuto, quando c'è il problema della scrittura come mezzo di comunicazione, con la necessità di rispettare l'ortografia delle parole.

DISLESSIA	DISORTOGRAFIA	DISCALCULIA	DISGRAFIA
De.Co.Ne (parole, non parole, brano di non parole)	DDE-2 (parole, non parole e frasi)	BDE-2	BHK
DDE-2 (parole e non parole)	DDO-2 (parole)	AC-MT 6-11	DGM-P
Prove di Lettura S. Lucia (parole, non parole, brano)	BVSCO-2 (frasi e brano)	AC-MT 11-14	
Prove MT per la Scuola Primaria (brano)	MT16-19 (parole e non parole)	ABCA	
Nuove prove MT per la Scuola Secondaria di I Grado (brano)	MT-Avanzate 3 (parole, non parole e brano)	ABCA 14-16	
MT-Avanzate 3 (parole, non parole e brano)		MT-Avanzate 3	
MT16-19 (parole, non parole e brano)			

Figura 1.1: Elenco test DSA.[\[6\]](#)

1.1.3 Test per Disgrafia e Disortografia

Sono proposti dalla letteratura un insieme di test appositi, ognuno mirato per uno specifico disturbo DSA. I test sono spesso somministrati agli alunni di età compresa tra i 4 e gli 11 anni, e valutano caratteristiche differenti, anche in base all'età dei soggetti.

Analisi dei test comuni (BHK and BVMGT)

Dal precedente capitolo si evince quali possono essere i comuni errori ortografici riguardanti lo scambio di lettere e gruppi di lettere all'interno di specifiche parole, mentre per gli errori grafici facciamo appello ad un insieme di parametri caratterizzanti più il tratto (il tracciato, il pattern) che la semantica.

Si intende con questo sottolineare come sia possibile dall'osservazione dei tratti (che chiameremo da qui in avanti **strokes**) studiare ed analizzare componenti propriamente tipiche sia in soggetti disgrafici tramite i parametri nella Sezione 1.1.1, che in soggetti disortografici analizzando il testo trascritto e quanto si avvicina ad un certo template originale (si fa riferimento alla Sezione 1.1.2).

Esistono, per una valutazione degli alunni, diversi strumenti che si basano sul comprendere se un determinato alunno sia più avanti o più indietro rispetto agli altri studenti della sua stessa sezione (classe scolastica di appartenenza).

I test più diffusi sono:

- BHK [7]- Scala sintetica per la valutazione della scrittura in età evolutiva.
- DGM-P [8]- Test per la valutazione delle difficoltà grafo-motorie e posturali della scrittura.
- BVSCO-2 [9]- Batteria per la Valutazione della Scrittura e della Competenza Ortografica. Stima le competenze del bambino nei tre aspetti della scrittura: grafismo, competenza ortografica e produzione del testo scritto.
- BVMGT [10]- Basato sulla riproduzione di disegni geometrici ed è largamente utilizzato in ambito internazionale per la valutazione di un insieme di abilità percettive, visuospatiali e cognitive in generale.

essi Di strumenti valutativi le produzioni grafiche e gli aspetti posturali degli alunni ne esistono ancora svariati (alcuni sono mostrati nella Figura 1.1), ma in questo documento ci soffermeremo ad analizzare solamente il primo e

l'ultimo sopra citati: **BHK**(test utilizzato in questa ricerca) e **BVMGT**(test di riferimento per sviluppi futuri).

BHK: caratteristiche del test

Per la valutazione della qualità della scrittura uno degli strumenti più conosciuti è la scala BHK degli studiosi olandesi Hamstra-Bletz e Blöte (1993). Hamstra-Bletz e Blöte ritengono che il normale percorso evolutivo della scrittura subisca delle modifiche durante il percorso di scolarizzazione. Una normale evoluzione non sembra essere presente nei bambini disgrafici che, pur avendo un grado di istruzione e di esercizio pratico adeguato, anziché produrre un'accettabile grafia, presentano un tracciato incerto, inadeguato rispetto alla forma e dimensione dei caratteri grafici. Hamstra-Bletz, Blöte hanno elaborato una griglia di valutazione della morfologia grafica dei bambini prendendo come riferimenti principali la qualità morfologica dei segni e la spaziatura dei grafemi [11].

Tale scala prende il nome di BHK ed analizza 13 parametri che permettono d'individuare le caratteristiche dell'atto grafico identificando l'eventuale presenza di disgrafia.

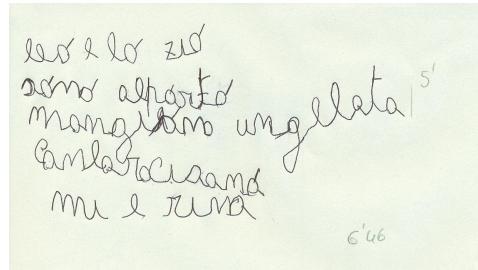
Tali parametri sono:

1. grandezza della scrittura;
2. margine sinistro non allineato;
3. andamento altalenante della linea di scrittura: lettere o parole mal allineate, i grafemi di una parola non sono allineati su una linea orizzontale, ma al di sopra o al di sotto di essa;

4. spazio insufficiente tra le parole: lo spazio tra i grafemi risulta essere talmente ridotto che il grafema terminale e quello iniziale di una parola risultano adiacenti o quasi uniti;
5. angoli acuti o collegamenti allungati tra le lettere;
6. collegamenti interrotti tra le lettere: pause nella traccia o assenza di collegamenti tra le lettere;
7. collisione tra le lettere a causa della ridotta distanza: i due grafemi risultano così tangenti o sovrapposti;
8. irregolarità della grandezza delle lettere: i grafemi risultano o troppo alti o troppo bassi rispetto alle dimensioni di riferimento (0,5 cm per i grafemi senza estensione e 1 cm per i grafemi con estensione);
9. misura incoerente tra lettere con e senza estensione in altezza;
10. lettere atipiche: deformazione delle lettere rispetto alla loro forma convenzionale;
11. forme ambigue delle lettere da un punto di vista morfologico così da poter essere confuse con altre;
12. lettere ritoccate o ricalcate non per correzione ortografica, ma per migliorarne la forma;
13. traccia instabile, scrittura incerta o tremolante.

Il BHK fu ideato per l'analisi dei parametri grafici di bambini olandesi e si è quindi presentata la necessità di adattare tale strumento alla popolazione italiana che presenta insegnamento e tipologia di codice linguistico differente. Viene utilizzato con i bambini tra il 2° anno di scuola primaria ed il 1°/2°

**Leo e lo Zio
sono al porto
mangiano un gelato
con loro ci sono
Mia e Rina**



(a) Testo template BHK. (b) Test eseguito da un bambino di 7 anni [11].

Figura 1.2: Esempio di applicazione del test BHK.

anno di scuola secondaria di primo grado e valuta la velocità (lento, adeguato, veloce) e la qualità di scrittura (disgrafico, non disgrafico). La prova consiste nella presentazione di un testo standard da copiare su un foglio bianco; il testo presentato è da copiare in corsivo, viene quindi chiesta al bambino anche la conoscenza di tale grafia. La complessità del testo è graduata: si parte da parole con configurazione sillabica semplice per proseguire poi con strutture sempre più complesse.

BVMGT: caratteristiche del test

Il test BVMGT (Bender Visual Motor Gestalt Test) è basato sulle figure di Trisciuzzi-Bender ed ha lo scopo di analizzare il livello di capacità percettiva -topologica o euclidea- in cui si trovano gli alunni all'inizio dell'apprendimento della lettura-scrittura [1]. Questa prova assicura la possibilità di conoscere la capacità di coordinamento oculo-manuale del bambino, ossia il suo livello topologico, permettendo sia di prevedere eventuali future difficoltà scolastiche, sia di prevenire futuri fallimenti.

Storicamente le figure vennero riprese dalla Bender negli anni trenta come prove di carattere medico. Queste nove figure vennero sottoposte ad adulti

e bambini, sia normali, sia insufficienti mentali o con turbe di personalità, al fine di evidenziare i caratteri di alcune malattie mentali. Bender chiamò questa prova « Visual Motor Gestalt Test ».

La prova mise in luce l'esistenza di una immaturità percettivo-motoria che può essere naturale, come pure patologica. Sono considerate naturali le incapacità di riprodurre le figure fino a una certa età, ossia se rientrano nello sviluppo del bambino, mentre se vengono in età in cui l'egocentrismo spaziale è da ritenersi concluso, rientrano nei quadri delle alterazioni e delle patologie.

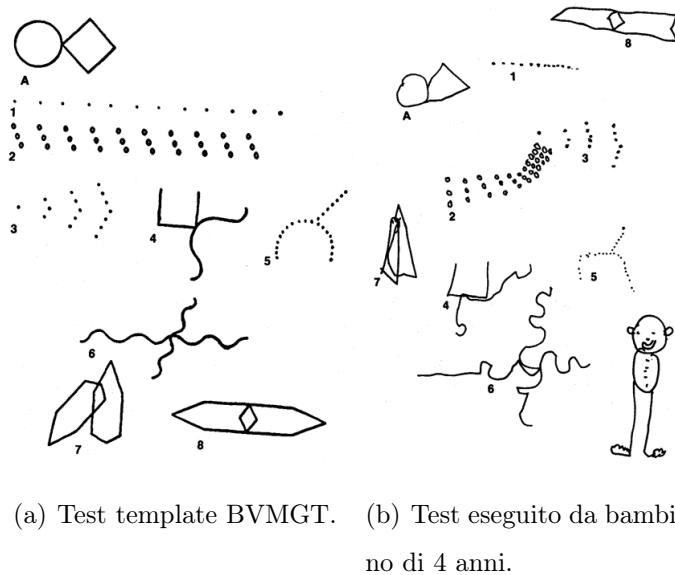
Esaminando i bambini che devono iniziare l'apprendimento della scrittura è possibile stabilire se l'organizzazione prattognosica è ancora a livello topologico oppure se si avvia a essere euclidea. Il sistema di verifica classico distingue quattro livelli strutturali, distinguendo nella riproduzione dei bambini:

struttura articolata : la copia è simile al modello e ne rappresenta gli elementi euclidei nelle angolature, nelle proporzioni delle parti, nella dimensione e nell'orientamento.

struttura disaggregata : la riproduzione è composta da elementi strutturati, ma disgiunti l'uno dall'altro, accostati l'uno a l'altro, a volte capovolti.

struttura disarticolata : la riproduzione è solo simile al modello, cioè si riconosce il modello ma non l'articolazione della struttura.

non-struttura : la riproduzione è al livello del grafismo spontaneo, dello sgorbio, dello scarabocchio, attribuibile a bambini al di sotto dei 4 anni.



(a) Test template BVMGT.

(b) Test eseguito da bambini di 4 anni.

Figura 1.3: Esempio di applicazione del test BVMGT [12].

Note

A differenza del test BHK, il BVMGT non verrà approfondito alla stessa maniera, non ne verrà data una forma algoritmica e tantomeno verrà somministrato a bambini. Questa scelta deriva dal fatto che il test non dispone di una metodologia di analisi oggettiva, non puo' essere schematizzato in forma tabellare con caratteristiche generali da osservare, ma bensì riguarda un analisi soggettiva. A meno che non si disponga di una mole molto ampia di dati, cosa che non risulta al momento in cui scrivo, una qualsiasi variante algoritmica per il test non porterebbe a risultati attesi.

1.2 Reti Neurali Artificiali

Le reti neurali e il deep learning sono argomenti di punta nelle nuove tecnologie e nell'informatica in generale, per la loro capacità di trovare migliori

soluzioni a problemi complessi come il riconoscimento di immagini, della voce e dei suoni, ed il processamento del linguaggio naturale. Introduciamo il concetto di rete neurale, discutendo riguardo alle differenze che vi sono fra una rete MLP (Multi Layer Perceptron) ed una rete convoluzionale CNN (Convolutional Neural Network), e rimandiamo al capitolo del progetto (Capitolo 2) l'analisi della loro implementazione e del loro utilizzo.

1.2.1 Reti Neurali artificiali

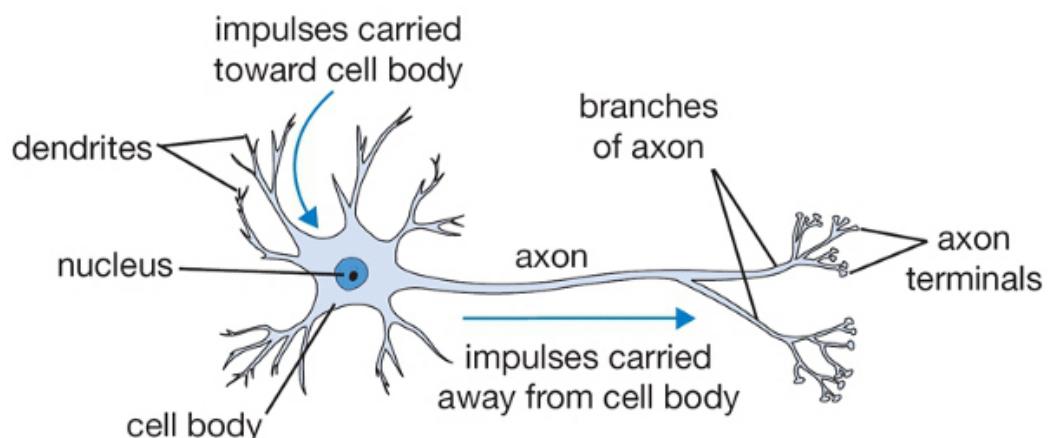
Una possibile definizione di Reti Neurali Artificiali, è stata fornita da Robert Hecht-Nielsen che si riferisce a questi oggetti come a: "... un sistema computerizzato composto da un largo numero di semplici elementi di processamento altamente interconnessi, che processano informazioni in base allo stato dinamico della loro risposta a degli specifici input esterni "[13].

In generale si può pensare ad una rete neurale come un modello computazionale che si ispira al modello biologico dei neuroni e che processa le informazioni replicando in qualche modo il metodo di processamento del cervello [14].

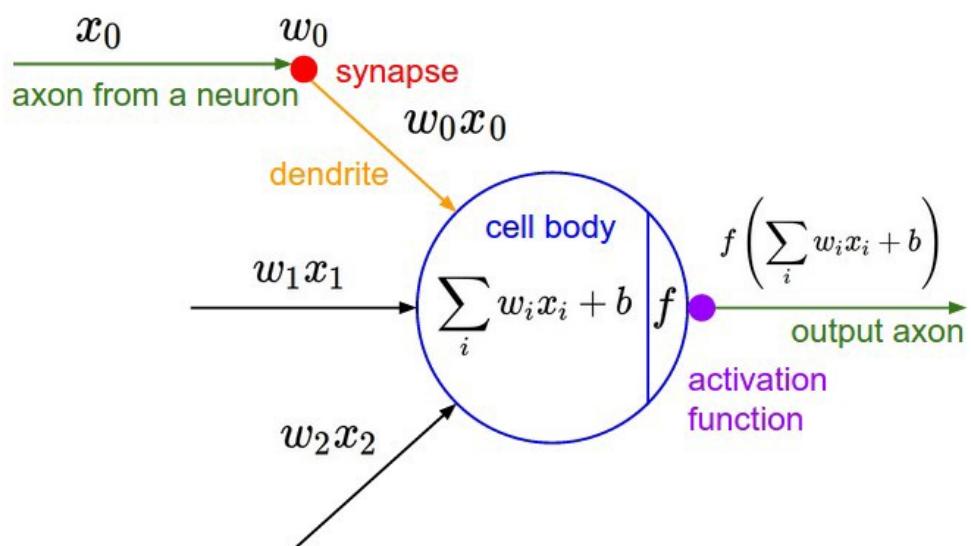
1.2.2 Motivazioni biologiche

L'unità base di processamento del cervello è il neurone. Si stima che nel sistema nervoso umano ci siano circa 86 miliardi di neuroni, interconnessi fra loro con $10^{14} \sim 10^{15}$ sinapsi [14].

L'unità base di elaborazione in una rete neurale artificiale è chiamata anche in questo caso neurone, ma spesso anche nodo o unità. Questa riceve degli input da alcuni neuroni, e processa un output per altri neuroni. Ogni input ha associato un peso (w), che viene assegnato in base all'importanza



(a) Neurone biologico.



(b) Neurone artificiale.

Figura 1.4: Rappresentazione di un neurone biologico ed uno artificiale.[15]

relativa rispetto agli altri input. Il nodo (o neurone) applica una funzione $f()$ ai valori pesati di input per riprodurre il valore in uscita.

L'idea è che l'importanza di una sinapsi (data dal valore del peso w) è da apprendere, e controlla l'influenza di quel collegamento sinaptico. Nel modello base, i dendriti mantengono il segnale nel corpo della cellula dove vengono sommati fra di loro. Se la somma supera una certa soglia, il neurone reagisce inviando un impulso lungo l'assone. Nel modello computazionale assumiamo che l'istante preciso in cui si inoltra il messaggio non sia di rilievo e che solo la frequenza degli impulsi comunichi le informazioni.

1.2.3 Architettura di una rete neurale

Dalla spiegazione precedente possiamo concludere che una rete neurale è costituita da neuroni. Biologicamente i neuroni sono collegati attraverso sinapsi dove fluiscono le informazioni (pesi per il modello computazionale). Quando addestriamo una rete neurale vogliamo che i neuroni si attivino ogni volta che imparano specifici modelli dai dati e modelliamo la velocità degli impulsi usando una funzione di attivazione. Non è ancora tutto, ci sono altri concetti da descrivere:

Nodi di input (livello di input): all'interno di questo livello non viene eseguito alcun calcolo, ma vengono semplicemente passate le informazioni al livello successivo (lo strato nascosto, il più delle volte). Un blocco di nodi allo stesso livello è anche chiamato layer.

Nodi nascosti (livello nascosto): l'elaborazione viene eseguita nei livelli nascosti, qui si eseguono i calcoli e si trasferiscono i segnali (o informazioni) al livello successivo, che può essere un altro livello nascosto

o il livello di output. È possibile avere una rete neurale senza strati nascosti, anche se questa configurazione non è molto utilizzata.

Nodi di output (livello di output): qui viene utilizzata una funzione di attivazione che restituisce il formato di output desiderato.

Connessioni e pesi : la rete è composta di connessioni, ogni connessione trasferisce l'uscita di un neurone j all'ingresso di un neurone i . In questo senso j è il predecessore di i e i è il successore di j , ad ogni connessione è assegnato un peso w_{ij} .

Funzione di attivazione : la funzione di attivazione di un nodo definisce l'output di quel nodo. Questo è simile al comportamento del perceptron lineare nelle reti neurali. Tuttavia, è la funzione di attivazione non lineare che aumenta il potere espressivo di tali reti.

Regola di apprendimento : la regola di apprendimento è un algoritmo che modifica i parametri della rete neurale, minimizzando l'errore che si commette nell'assegnare una certa label all'ingresso. Viene modelato un problema di ottimizzazione che ha come variabili i pesi e le soglie della rete, e come funzione obiettivo la funzione di errore (loss function). L'algoritmo più usato per l'aggiornamento dei parametri è il Backpropagation.

1.2.4 Tipi di reti neurali

Ci sono molte classi di reti neurali e molte sottoclassi per ognuna di queste. Ci concentreremo sui due tipi di Reti Neurali Feedforward utilizzate nel progetto: Percettrone multi-livello, e Rete Neurale Convoluzionale.

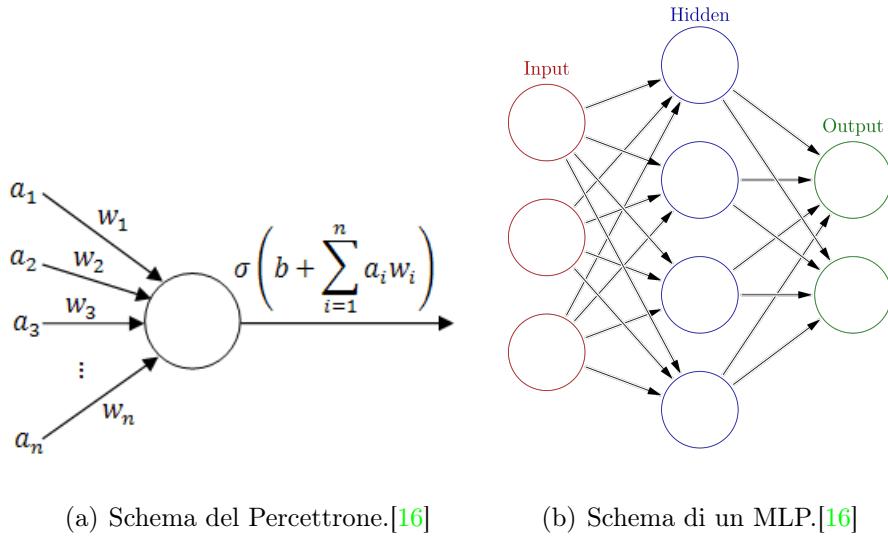


Figura 1.5: Percettrone vs. Percettrone multistrato.

Reti Neurali Feedforward

Una rete neurale feedforward è una rete neurale artificiale in cui le connessioni tra le unità non formano un ciclo. In questa rete, le informazioni si spostano solo in una direzione, in avanti, dai nodi di input, attraverso i nodi nascosti (se presenti) fino ai nodi di output.

Percettrone e MLP : Il Percettrone è il tipo di rete neurale più semplice e non contiene alcun livello nascosto, questo significa che è composto da un solo livello di nodi di uscita. In un perceptron singolo, viene moltiplicato scalarmente il livello di input con i pesi, ed applicata sul risultato la funzione di attivazione. Una rete MLP (multi-layer perceptron, Percettroni multistrato) invece consiste di più livelli di computazione, connessi fra loro tipicamente in avanti (feedforward). Ogni neurone di ogni livello è connesso con ognuno del livello successivo. Queste reti vengono chiamate completamente-connesse (fully-connected). Tipicamente viene scelta come funzione di attivazione la funzione sigmoide,

che ha la particolarità di essere non-lineare e facilmente derivabile. Ed è proprio la sua non-linearità che ci permette di avere delle ottime prestazioni (anche in problemi non linearmente separabili), con un numero così piccolo di neuroni nella rete.

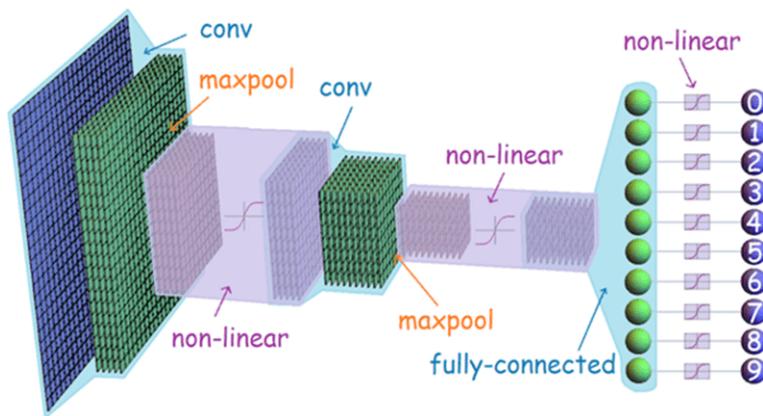


Figura 1.6: CNN per riconoscimento di caratteri numerici [17].

Reti Neurali Convoluzionali : Le Reti Neurali Convoluzionali sono costruite secondo il pattern organizzativo della corteccia visiva, le unità rispondono agli stimoli in una regione ristretta dello spazio conosciuta come campo recettivo (receptive field). I campi ricettivi si sovrappongono parzialmente, coprendo l'intero campo visivo. La risposta dell'unità può essere approssimata matematicamente mediante un'operazione di convoluzione. Queste reti sono una variazione dei Percetroni multi-strato, poiché gli strati convoluzionali possono essere visti come un preprocessing per gli strati finali fully-connected di una CNN. Le loro ampie applicazioni riguardano il riconoscimento di immagini e video, sistemi di raccomandazione e l'elaborazione del linguaggio naturale.

1.3 Background Analisi della scrittura

La scrittura manuale ed i vari testi prodotti a mano (Handwriting) continuano a persistere come mezzo di comunicazione e memorizzazione di informazioni giorno dopo giorno, nonostante l'introduzione pervasiva delle tecnologie moderne [18]. Data la sua ubiquità nei rapporti umani, il riconoscimento automatico di manoscritti ha un forte impatto pratico, a partire dalla verifica delle firme (signature verification [19]) fino alla traduzione di manoscritti per consultazione online (google books project [20]).

Le tecniche di riconoscimento ed analisi di testo manoscritto (Handwriting Recognition, HR) coprono un'area interessante del riconoscimento di forme. I moderni sistemi di HR sono composti da stadi consecutivi quali: preprocessing, feature extraction, classification e riconoscimento (nel preprocessing si considerano anche tecniche di normalization e segmentation). Una fase particolarmente importante è l'estrazione delle feature, le quali influenzano positivamente o negativamente le performance del modello [21]. La selezione di un insieme appropriato di feature per rappresentare gli esempi di input è il principale problema di questi modelli e la scelta della tecnica migliore è basata sulla minimizzazione dell'errore di classificazione.

Si capisce come la scelta delle feature condiziona la scelta delle tecnologie da usare, e ne determina la riuscita di una eventuale classificazione (predizione di un attributo categorico) o regressione (predizione di un attributo numerico).

Ci sono varie tecniche per l'analisi della scrittura manoscritta, che si differenziano in base a due diverse tecniche di estrazione dei dati: online e offline.

1.3.1 Scrittura Online ed Offline

Nell’analisi di scrittura online entrano in gioco spesso delle tecnologie esterne come tavole digitali, videocamere, sensoristica di comunicazione fra la penna ed tablet. Anche la smart-pen ed il tablet Wacom utilizzati per questo progetto appartengono a questa categoria. Nell’ estrazione offline, si parla invece di scansione dei documenti con un’acquisizione degli stessi meno dettagliata e priva di certi tipi di informazioni che potrebbero essere utili per il riconoscimento del testo. Infatti una delle differenze più grosse che ci sono fra l’analisi online ed offline della scrittura è la presenza di una cronologicità dei tratti e dei punti che compongono i tratti. Questa cronologicità può fornire informazioni importanti sullo scrittore, sul modo di scrivere alcune parole e dunque sulle parole che risultano più semplici o più difficili da scrivere. Esistono molti algoritmi per l’analisi ed il riconoscimento di testi manoscritti online, dove il workflow spesso comprende l’utilizzo di HMM (Hidden Markov Model, modelli Markoviani nascosti) per poter caratterizzare mediante l’utilizzo di questi modelli stocastici la presenza di un pattern specifico di un carattere, piuttosto che di un altro; ed in base a queste probabilità modellare l’evento che si abbia una certa parola (vista come congiunzione di eventi “carattere”) piuttosto che un’altra. Ovviamente le feature corrispondono ai punti registrati dal tablet e dalla penna elettronica.

D’altro canto, nonostante il riconoscimento di manoscritti offline sia un problema più complesso (non disponendo di molti dati, se non il valore di ogni pixel del documento), si sono sviluppate tecniche di estrazione di feature molto sofisticate. Queste partono da una vasta gamma di algoritmi per il pre-processing dei documenti, l’analisi e l’estrazione delle feature, fino alla vera e propria componente di riconoscimento. Fanno parte del bagaglio di studi algoritmi di filtering come la binarizzazione, il miglioramento del

segna e la riduzione del rumore; algoritmi per l'estrazione e l'analisi dei contorni e dello scheletro; algoritmi riguardo la segmentazione degli oggetti ed il loro raggruppamento in componenti connesse e pixel adiacenti; algoritmi che compiono operazioni di dilatazione ed erosione delle componenti connesse all'interno dei documenti stessi, ed algoritmi più usati come il RLSA e la trasformata di Hough per trovare le linee all'interno della pagina per caratterizzare lo skew, ed eventualmente migliorare la fase di pre-processing.

Poichè il nostro obiettivo non riguarda il riconoscimento del testo, ma piuttosto una buona estrazione di feature, vengono scelte nel corso del progetto sia il mantenimento dei dati sotto forma di online handwriting, che la loro trasformazione in immagini, cioè offline handwriting.

1.4 Background Tecnologie

Il digitalizzatore (digitizer) è un dispositivo che permette di riconoscere la posizione della penna o delle dita su uno schermo, consentendoci di utilizzare anche l'inchiostro digitale (digital ink). In base al tipo di digitalizzatore, questo può trovarsi sopra o sotto lo schermo [22].

Esistono due grandi classi di digitalizzatori: quelli passivi, che non inviano dati verso l'esterno, ma ricevono solamente le sollecitazioni esterne, e quelli attivi, che inviano e ricevono segnali. I principali digitalizzatori passivi sono quelli resistivi e capacitivi; mentre tra i digitalizzatori attivi principali vi sono quelli elettromagnetici, come quelli prodotti dalla giapponese Wacom, e quelli elettromagnetici/capacitivi dalla israeliana N-Trig.

Nel nostro progetto utilizzeremo la tecnologia Wacom, con il tablet e la smart-pen Bamboo, che ci permettono di ottenere l'inchiostro digitale della scrittura.

1.4.1 I Digitalizzatori Passivi

I digitalizzatori passivi sono un tipo di digitalizzatore che non invia dati verso l'esterno, ma riceve solamente delle sollecitazioni esterne. In questa classe di dispositivi possiamo distinguere le tipologie di resistivo e di capacitivo.

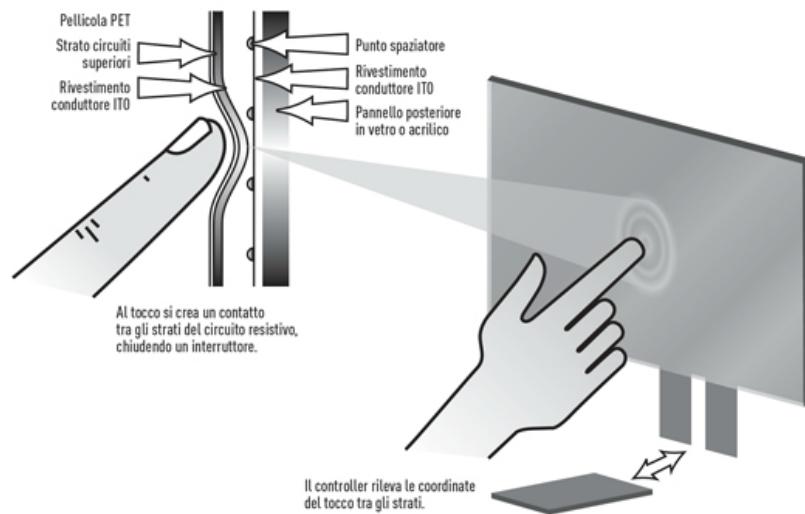
Digitalizzatore passivo resistivo

Il digitalizzatore passivo resistivo è uno dei primi utilizzati come “touch-screen”. Questo è sensibile alla pressione sullo schermo, indipendentemente dall'oggetto che viene utilizzato. A seconda della qualità dello schermo, può essere capace o meno di riconoscere la pressione esercitata, ma in nessun caso riesce a rilevare oggetti in prossimità dello schermo.

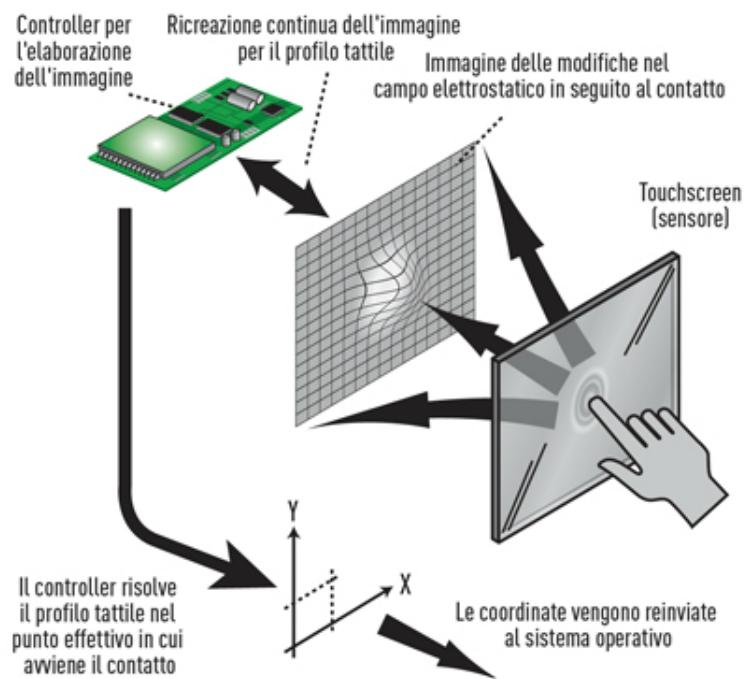
Da un punto di vista fisico il digitalizzatore passivo resistivo è composto da due strati di materiale conduttivo che, nel momento in cui un oggetto viene premuto sullo schermo, entrano in contatto riuscendo a determinare la posizione dell'oggetto. I digitalizzatori passivi resistivi variano in base alla sensibilità che hanno alla pressione, inoltre nello scrivere è importante non appoggiare sullo schermo altre parti non comprese nella scrittura, come il polso, la mano per tenere stabile il foglio, ecc. Questa tecnologia non è inoltre in grado di riconoscere la posizione in modo preciso (inferiore al millimetro).

Digitalizzatore passivo capacitivo

Il digitalizzatore passivo capacitivo è, al momento, il più diffuso dei digitalizzatori ed è installato sulla quasi totalità dei Tablet PC e smartphone. Posto sopra il pannello LCD, questo digitalizzatore riconosce la presenza di oggetti che sfiorano lo schermo, o che vengono premuti contro di esso, utilizzando la misura della loro capacità elettrica. Poggiando, o anche solo



(a) Schermo resistivo.



(b) Schermo capacitivo.

Figura 1.7: Schema di funzionamento schermi passivi.[5]

avvicinando, l'oggetto sullo schermo questo fa cambiare la capacità tra due singoli conduttori di una griglia, posta al di sotto dello schermo; infine dei sensori rilevano il cambiamento del campo magnetico, e determinano la posizione precisa dell'oggetto. Pensato più che altro per l'input mediante dita, il digitalizzatore capacitivo è capace di riconoscere la posizione della penna solo con una precisione modesta in quanto per operare richiede una superficie di contatto abbastanza vasta (in genere una circonferenza di almeno mezzo centimetro di diametro).

Per scrivere su un digitalizzatore capacitivo è quindi necessario utilizzare penne dalla punta molto larga come la Wacom Bamboo Stylus; si tratta inoltre di un digitalizzatore che non è in grado di riconoscere la pressione esercitata se non in modo indiretto ed estremamente poco preciso.

Il digitalizzatore passivo capacitivo non può essere usato con tutti gli oggetti: i materiali isolanti, o anche semplicemente le dita coperte da guanti, non sono infatti percepiti.

1.4.2 I Digitalizzatori Attivi

I digitalizzatori attivi hanno la capacità di inviare e ricevere segnali, a differenza di quelli passivi.

Digitalizzatore attivo Wacom Penabled

Il digitalizzatore attivo elettromagnetico Wacom Penabled è composto da una griglia che crea un piccolo campo elettromagnetico verso l'esterno e da una particolare penna che si attiva con questo campo. Tale digitalizzatore viene montato dietro il pannello LCD ed offre una notevole precisione di scrittura (a parte sui bordi, dove c'è un calo di precisione) unita ad una sensibilità a 256 livelli di pressione. Il digitalizzatore è sensibile alla sola

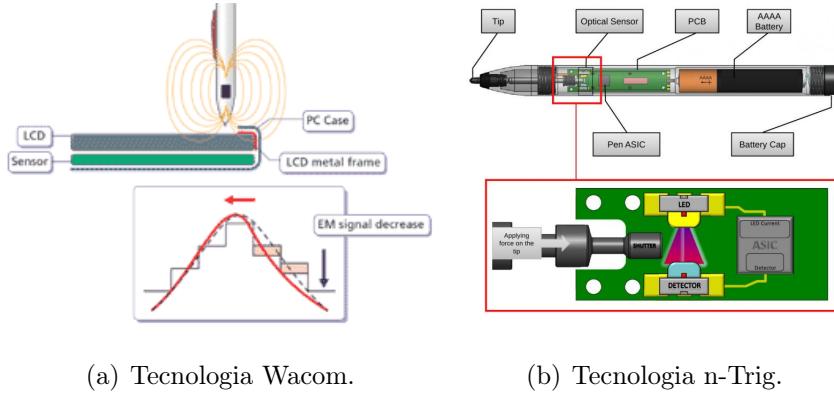


Figura 1.8: Schema di funzionamento tecnologie attive.[23]

penna e non alle dita: si può quindi appoggiare il polso senza problemi sullo schermo e scrivere in modo completamente naturale, ma non si possono usare in alcun modo dita o oggetti diversi dalla penna Wacom. Essendo montato dietro il pannello LCD, non interferisce in alcun modo con la qualità dell'immagine.

Con l'aumentato interesse nei confronti del multitocco nella maggior parte dei Tablet PC il digitalizzatore Wacom è stato affiancato da un digitalizzatore passivo capacitivo, in modo che il Tablet PC possa essere usato con la penna e con le dita: in caso sia presente il doppio pannello il digitalizzatore capacitivo viene però disattivato nel momento in cui la penna si avvicina allo schermo, permettendo al polso di appoggiarsi ad esso senza problemi nel momento in cui si scrive con la penna.

Capitolo 2

Il Progetto

L'obiettivo di questo progetto è quello di determinare dalla scrittura di un alunno se questo possiede le caratteristiche di un soggetto DSA o meno. Dalla conoscenza appresa nel precedente capitolo (Capitolo 1) in ambito di DSA, analisi della scrittura, principi di reti neurali e funzionamento delle nuove tecnologie per la scrittura digitale e il digital Ink, il progetto si struttura a partire dalla scelta del materiale: un tablet slate large ed una penna digitale Wacom (entrambi di famiglia Bamboo). Chi lavora in questo settore ha provato negli anni passati a implementare algoritmamente un'analizzatore di testo, ma vengono usate tecniche e linguaggi differenti da quelli qui proposti (nel corso dello sviluppo del prototipo sono venuto a conoscenza di implementazioni in MATLAB, delle quali non risulta esistere nessuna pubblicazione).

Nella pianificazione delle attività è stato necessario strutturare il progetto in cinque fasi principali. La prima riguarda un'analisi del formato dei dati (prodotto dalla tavola Wacom) per poter estrarre la componente significativa dagli stessi (nonchè il file che conteneva le informazioni della scrittura). Una successiva fase ha come scopo decodificare tali informazioni ed ottenere i

punti per i quali la penna ha lasciato l'inchiostro digitale (digital Ink). Si procede implementando algoritmamente l'analisi della scrittura basandoci sulla struttura di analisi utilizzata nel test BHK. Questa fase ha richiesto un impegno maggiore poichè il problema di suddivisione dei punti in stroke, ed il raggruppamento degli stroke in parole e successivamente in righe, è uno degli argomenti di punta quando si parla di HWR (nel nostro caso HWAnalysis). Anche la successiva fase di creazione del dataset è stata molto impegnativa poichè non si trovano online dataset con le caratteristiche di nostro interesse (scrittura di bambini secondo il test BHK). È stato necessario apportare delle modifiche all'idea di partenza e dunque costruire un dataset *ad hoc* fittizio composto dalle solite 5 righe iniziali del test BHK, ma piuttosto che avere la scrittura di bambini di età fra i 5 e gli 11 anni, abbiamo preferito raccogliere inizialmente i dati dalla mano principale e secondaria di studenti universitari e non, simulando con la scrittura della mano secondaria una forte disgrafia. Nell'ultima parte del progetto andremo nella scuola media di Scarperia e San Piero a prelevare i dati veri dai bambini delle classi 1°; il primo dataset fittizio ci ha consentito di fare i primi test. L'ultima fase riguarda l'implementazione di più reti neurali, per diversi tipi di feature estratte, in modo da poter fare un confronto fra le varie tecniche e poter scegliere quella che classifica meglio i dati in ingresso.

Vediamo più nel dettaglio le fasi sopra elencate, cosa nello specifico è stato fatto, quali problemi si sono riscontrati, e come sono stati superati.

2.1 Estrazione dati

Vengono scelti come strumenti di generazione dei tratti il tablet slate e la penna digitale prodotti da Bamboo, tecnologia Wacom, che oltre ad

essere una buona tecnologia, molto precisa per quanto riguarda il digital Ink, ci consente mediante la loro piattaforma di scaricare i documenti digitali prodotti. Le estensioni dei file digitali da scaricare possono essere molteplici, in base alla necessità di avere dati più o meno elaborati (WILL, SVG, PNG, PDF, ecc..). Quello che a noi interessa è ottenere i punti per i quali la penna è passata nella fase di scrittura. Dunque l'estensione che fa al caso nostro è quella di tipo SVG. Si nota che l'SVG prodotto dalla piattaforma restituisce il bordo dei tratti, e non i tratti in generale o lo scheletro degli stessi.

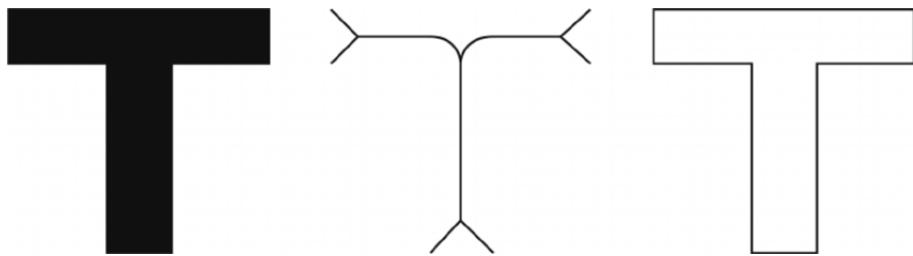


Figura 2.1: Esempio di scheletro e di contorno [24].

E poichè non esistono tecniche che ci permettono di risalire allo scheletro di un'immagine partendo dai contorni, l'unica strada da intraprendere risulta essere quella di utilizzare il documento WILL (tipo di file paragonabile ad un file ZIP) strutturato come alla Figura 2.2, e a partire da questo produrre noi stessi il file SVG adeguato (caratterizzando la scrittura con il suo scheletro). All'interno del file WILL, oltre ai file di estensione nota (XML, JPEG, SVG), c'è un file binario (.protobuf) contenente tutte le coordinate dei punti per i quali è passata la penna digitale, ed ha lasciato l'inchiostro *Ink*. Questo è il 'main-file', contiene tutti i dati che a noi interessa decodificare e questa sezione tratterà del procedimento con il quale si sono ottenuti gli strokes a partire dal file WILL.

2.1.1 Formato file WILL

Principalmente i dati WILL rappresentano gli strokes della penna come digital Ink. Uno stroke prodotto da una penna consiste in una coppia di punti posizione (x e y), e per fornire una qualità migliore del tratto, le penne digitali memorizzano informazioni aggiuntive come il colore, lo spessore e la forma dello stroke. Il file WILL è impacchettato usando uno standard ISO chiamato OPC (Open Packaging Convention), ciò che dobbiamo estrarre è un file di tipo SVG, linguaggio basato sull' XML per descrivere grafica vettoriale. Le informazioni SVG sono codificate in binario usando lo standard *protocol buffer* di Google, che fornisce una compatibilità multipiattaforma molto rapida in codifica e decodifica.

```

Example.will
|
└─ [Content_Types].xml
└─ props
   ├─ app.xml
   └─ core.xml
└─ sections
   ├─ media
   │   ├─ image2049281480.jpeg
   │   └─ strokes387757975.protobuf
   ├─ section0.svg
   └─ _rels
      └─ section0.svg.rels
└─ _rels
   └─ .rels

```

Figura 2.2: Albero composizione file WILL.

2.1.2 Formato dati WILL

I dati memorizzati all'interno del file WILL (specificatamente nel file protobuf) sono di tipo *Path*, oggetto composto come in Figura 2.3.

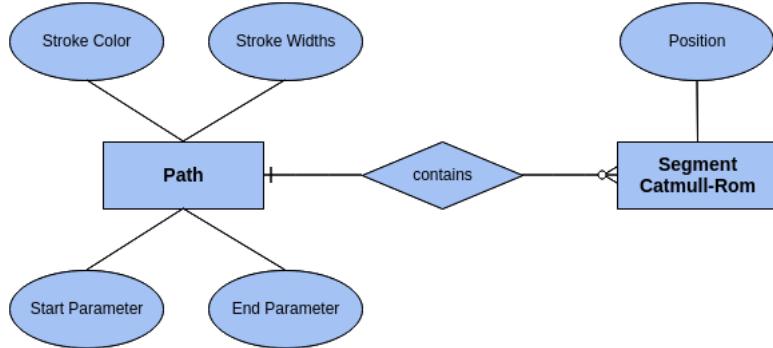
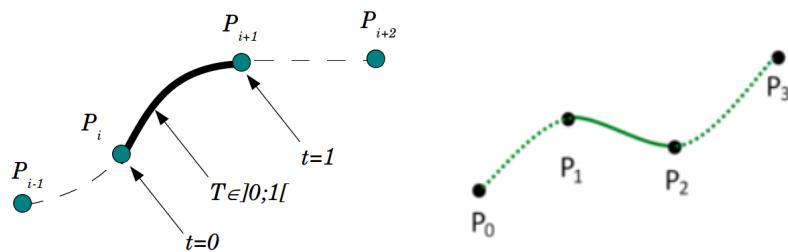


Figura 2.3: Modello concettuale dell’oggetto Path

Path

L’entità Path è il blocco fondamentale in digital Ink. Ogni stroke è rappresentato come un’istanza di Path. Gli attributi principali oltre alle coordinate sono lo spessore (width) ed il colore (color, che assumeremo sempre come black). Gli attributi Start Parameter e End Parameter specificano il valore t dell’ultimo segmento dove il path è iniziato e terminato.



(a) Geometria dell’oggetto Path

(b) Segmento Catmull-Rom

Figura 2.4: Oggetto Path e algoritmo Catmull-Rom

Per le coordinate invece viene usato l'algoritmo Catmull-Rom che interpola i punti P_{i-1} , P_i , P_{i+1} , P_{i+2} per definire il tratto di curva fra P_i e P_{i+1} .

Codifica e decodifica

Il processo di **codifica** è suddiviso in due step:

Step 1. Convertire i valori floating-point in interi usando una precisione specificata dallo sviluppatore:

```
integerValue = floatValue * (10 ^ decimalPrecision);
```

Step 2. Applicare la codifica delta encoding sui valori interi:

```
encodedValues[0] = integerValues[0];
for (i = 1; i < n; i++)
{
    encodedValues[i] = integerValues[i]
        - integerValues[i - 1];
}
```

Per ottenere i valori originali, il processo di **decodifica** è opposto, e dunque applicare la decodifica delta encoding per riottenere valori interi, ed in seguito convertire i valori interi in floating-point usando la stessa precisione sopra specificata.

2.1.3 Decodifica file protobuf

Vediamo come poter ottenere delle istanze della classe *Path* a partire dalla definizione della classe come in Figura 2.3, dal file binario *strokes.protobuf* e dalle specifiche in Figura 2.5.

Description	Bytes sequence
128 bit varint	<i>Path 1 message length</i>
Bytes	<i>Path 1 message bytes</i>
128 bit varint	<i>Path 2 message length</i>
Bytes	<i>Path 2 message bytes</i>
...	...
128 bit varint	<i>Path N message length</i>
Bytes	<i>Path N message bytes</i>

Figura 2.5: Lista a lughezza variabile degli stroke codificati

Nella Figura 2.5 si capisce come poter estrarre i bit necessari che rappresentano un *Path* da questo tipo di file binario, e dunque proseguire poi a leggere i primi 128 bit dal file, tradurli nel valore intero di bit che rappresentano (supponiamo che nei primi 128 bit ci sia scritta la codifica binaria dell'intero x), prelevare i successivi x Byte dal file binario sapendo che questi rappresentano un oggetto di tipo *Path*, ed ottenere lo stroke convertendo gli x Byte secondo le direttive della documentazione protocol buffer, mediante la struttura della classe descritta in [25]. Per convertire ognuno degli stroke è necessario iterare il procedimento a partire dal primo bit successivo.

Questo, se vengono tradotti adeguatamente i Byte in istanza di oggetti *Path*, ci permette di ottenere una lista di strokes ordinata in base alla loro creazione, che utilizziamo per tradurli in oggetti *Polyline* dello standard SVG.

Generiamo l'intestazione del file SVG (la componente *Meta* del file, cioè larghezza e altezza del foglio, ecc..) in base alle dimensioni del foglio virtuale associato al tablet Bamboo, ed aggiungiamo le *Polylines* al file.

Fatto questo abbiamo un file SVG con il quale incominciare a lavorare, rappresentante i punti che ricreano lo Sketch sul foglio.

2.1.4 Esempio utilizzo

Il primo passo da fare per la decodifica è quello di scaricare i file .will dalla piattaforma o dall'app mobile [26]. Per fare ciò seguire le istruzioni alla pagina ufficiale di Wacom Inkspace , e cioè:

Fase 1. Scarica l'app Wacom Inkspace.

Fase 2. Segui le istruzioni passo per passo per configurare e abbinare il tuo dispositivo Wacom con l'app.

Fase 3. Quando ti viene richiesto, registrati o accedi con il tuo Wacom ID per attivare Inkspace.

Fase 4. Accedi a Inkspacee scarica il formato .WILL [27].

A questo punto è necessario configurare l'ambiente per eseguire il progetto, e quindi oltre a clonare la repository da github [28] e seguire i comandi disponibili nei README.md, sarà necessario scaricare da Inkspace i file in due estensioni differenti (per ogni file):

.WILL da posizionare nella cartella '_wills'.

.DOC da posizionare nella cartella '_docs'.

Fatto questo è possibile utilizzare il progetto OpenSource **will_to_svg** (adeguatamente modificato ed inglobato nella repository sopra linkata) con

il codice per la decodifica di ogni file WILL nella cartella `_wills` in file SVG, posizionandoli nella cartella apposita `_svgs`.

2.2 Elaborazione degli stroke

Nella Capitolo 1 si sono affrontati argomenti riguardo allo studio e approfondimento sui più comuni test per DSA, concentrando sul BHK e BVMGT. In questa Sezione andremo a richiamare le nozioni della Sezione 2.1 (in particolare 2.1.3) e riprenderemo il filo dagli oggetti *Path* decodificati dal file binario `.protobuf` in un file SVG. Di seguito, studieremo come sono rappresentati gli strokes, come si procede al loro raggruppamento per la creazione di parole e successivamente di righe. In fine analizzeremo quali sono i parametri del test BHK che possiamo valutare nel testo, come decidiamo di implementare questa valutazione e quali di questi trascureremo, motivando anche la nostra scelta.

2.2.1 Rappresentazione degli stroke

Dal momento in cui disponiamo degli stroke come oggetti SVG di tipo *Polyline*, possiamo iniziare il nostro procedimento di analisi degli stroke prelevando da questi alcuni parametri di nostro interesse. Ogni stroke rappresenta un tratto di Ink in cui il soggetto scrittore non ha alzato la penna dal foglio. Ognuno di questi tratti è formato da un insieme di coordinate spaziali, e per ognuna di queste vi sono memorizzati colore e spessore del tratto. Il primo passo sta nel prelevare l'insieme di coordinate (x, y) di ogni punto della *Polyline*, ed i parametri che indicano il colore e la larghezza dello stroke. L'ordine degli stroke è una caratteristica importante che non deve

essere persa nell'elaborazione delle informazioni. Utilizzeremo quindi una struttura FIFO (una lista) per tenerne di conto.

2.2.2 Creazione Parole e Righe dagli stroke

Si procede alla creazione delle parole contornando ogni stroke di un rettangolo (il rettangolo circoscritto), e creando le parole in base agli stroke i cui rettangoli si intersecano, anche per una piccola percentuale di area.

Al termine di questo procedimento, si controlla che le parole siano completamente disgiunte, ed in caso contrario si uniscono due parole che si intersecano ipern modo da formarne una sola.

Questa ultima parte serve per accorpare insieme le parole, e gli stroke in generale, che sono nella stessa area di spazio nel foglio. Può infatti succedere che durante la scrittura si esegua una correzione ad una parola precedentemente scritta: con questa accortezza vengono fuse le parole alle correzioni. In Figura 2.6 si mostrano le parole ottenute al termine dell'accorpamento degli stroke e delle parole ritoccate.

A questo punto si prelevano le parole ad una ad una, e si controlla la loro posizione reciproca per verificare che siano nella stessa riga o meno. Questo viene fatto sulla base della posizione rispettiva fra le parole, relativa all'asse delle x . Infatti se una parola viene dopo ad un'altra, è logico pensare che si trovi anche in una posizione successiva riferita all'asse delle ascisse (x).

Vengono quindi create le righe, sperabilmente cinque, riguardanti il testo del test BHK. In Figura 2.7 viene mostrata l'identificazione con successo di una riga ottenuta fondendo le parole assieme.



Figura 2.6: Creazione delle parole unendo gli stroke



Figura 2.7: Creazione delle righe unendo le parole consecutive

2.2.3 Riconoscimento del testo

Come specificato nella Sezione 2.1.4 per ogni file è necessario scaricare, oltre al WILL, un file di estensione DOC. Questo file contiene in ASCII ciò che gli algoritmi della piattaforma InkSpace di Wacom hanno riconosciuto dal testo scritto mediante i loro supporti. Infatti per questo progetto viene delegato a terze parti di riconoscere il testo, e viene usato come parametro di valutazione per il test BHK anche la traduzione, che riteniamo affidabile, fatta dalla piattaforma Wacom.

Quello che viene fatto da parte nostra nel progetto è trasformare i file DOC in TXT, leggere ogni parola in ogni riga e confrontare la parola del template (prelevata dal file *template.txt* sul quale è trascritto il testo che i bambini devono ricopiare) con la rispettiva parola riconosciuta da InkSpace di Wacom. Analizzeremo poi le differenze in base ad alcuni parametri che approfondiremo nella Sezione 2.2.4.

2.2.4 Interpretazione del test BHK

Questa sezione del documento usa i parametri visti alla Sezione 1.1.3 del Capitolo 1, per definire dei metodi che siano capaci di analizzare gli stroke in modo più simile possibile a come viene fatto dal test BHK. Durante la fase di creazione delle parole e delle righe, descritta nella Sezione 2.2.2, vengono fatte sugli stroke un insieme di operazioni per estrarre le 13 caratteristiche del test. Purtroppo alcune di queste non verranno considerate per alcune difficoltà nel processamento, che comunque descriveremo successivamente. Vediamo insieme quali caratteristiche preleviamo e come.

Grandezza della scrittura : Il valore della grandezza della scrittura viene prelevato a partire dalle righe già correttamente formate. Per ogni parola si evidenziano i punti di minimo e massimo locali, osservando per ogni punto una finestra di 5 valori (2 precedenti e 2 successivi). Trovati i minimi locali, questi vanno a formare (uniti per ogni parola di una specifica riga) i minimi locali di ogni riga. Si costruisce una retta di miglior approssimazione (ai minimi quadrati) per i punti di massimo e minimo della riga, e si valutano le distanze medie fra queste due rette, in ogni riga. Viene fatta la media delle distanze medie, e si calcolano così i pixel relativi all'altezza media della scrittura.



Figura 2.8: Grandezza della scrittura

Dovrebbe essere omogenea la distanza fra la retta superiore arancio e quella inferiore rossa.

Margine sinistro non allineato : L'allineamento del margine sinistro viene fatto a partire dalle due rette precedentemente costruite. Nel margine sinistro di ogni riga viene calcolato un punto: il punto medio fra la retta di approssimazione dei massimi, e quella di approssimazione dei minimi. Viene poi costruita la retta di miglior approssimazione dei punti (di colore viola nella Figura 2.9) la cui pendenza, calcolata come y/x , indica di quanto il margine sinistro non è allineato.

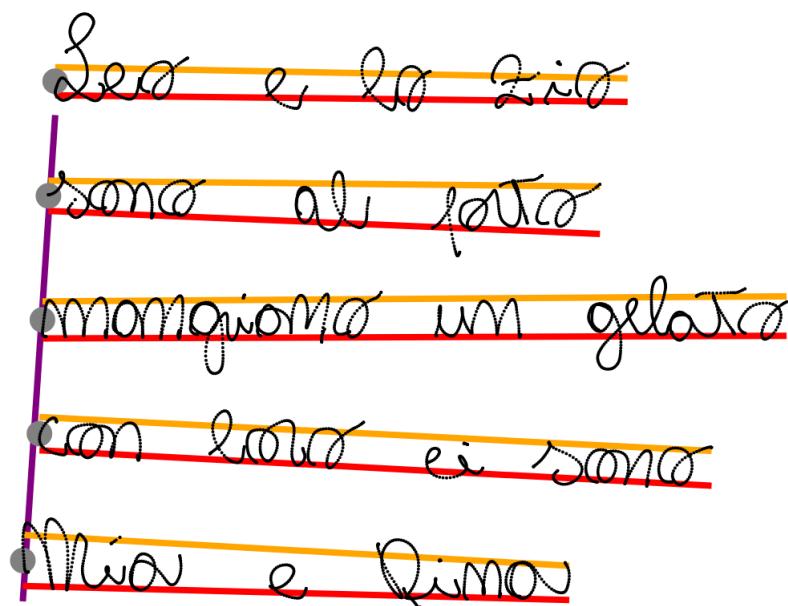


Figura 2.9: Margine sinistro non allineato

In grigio i punti, ed in viola la retta del margine sinistro.

Andamento altalenante della linea di scrittura : Per ogni parola abbiamo già trovato i minimi ed i massimi locali. Prelevando i minimi locali e provando, all'interno di ogni parola, a tracciare la retta di miglior approssimazione, troviamo la pendenza della parola specifica. Questa pendenza, in composizione con la pendenza delle altre parole all'interno

della riga, evidenzia come queste siano in qualche modo coordinate fra loro o meno. Infatti, per ottenere informazioni sull'andamento altalenante della scrittura, si procede sommando le distanze che vi sono fra le pendenze delle parole stesse con la pendenza della riga.

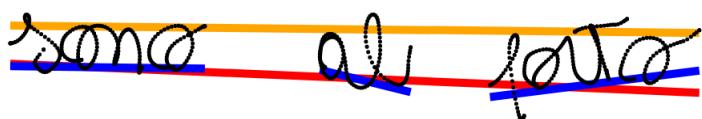


Figura 2.10: Andamento altalenante della linea di scrittura

Notare la pendenza delle parole rispetto a quello della riga.

Spazio insufficiente tra parole e lettere : Nella creazione delle parole e successivamente delle righe vengono sì, uniti gli stroke i quali rettangoli circostritti collidono, ma si va ad unire anche quelli che hanno una distanza fra loro inferiore ad una certa soglia (8 pixel verticali e 10 orizzontali). Questo perché nello scrivere una parola relativamente lunga in corsivo si lascia la penna a metà, per proseguire accanto. Per esempio nello scrivere la parola del test *mangiano* viene spesso scritto *man* e successivamente *giano*. Questo porterebbe a non vederle come un'unica parola, nonostante lo siano. Dunque le parole vengono unite per formarne una sola, ma si memorizza che è avvenuto questo tipo di artefatto. Lo stesso procedimento viene fatto nel verso opposto: poiché nel nostro problema sappiamo precisamente quante parole devono presentarsi per ogni riga, ci si ferma quando si trovano in ciascuna riga il numero di parole esatte. Questo può portare a definire uno spazio fra le parole in una riga differente rispetto ad un'altra, o addirittura

in una stessa riga uno spazio differente fra parole diverse. Si annota anche se questo tipo di irregolarità fra le parole.



Figura 2.11: Spazio insufficiente tra le parole e tra le lettere

Con colori diversi elementi troppo distanti che appartengono alla stessa parola.

Lettere ritoccate o ricalcate : Come specificato quando si parlava di unire le parole in base anche a delle correzioni (Sezione 2.2.2) ogni qualvolta si torna indietro per ritoccare una lettera o una parola, questo viene rilevato, viene associata la correzione alla parola e inserito lo stroke in quella parola. Quando questo avviene se ne tiene traccia per poterlo valutare nel test.



Figura 2.12: Lettere ritoccate o ricalcate

In rosso il tratto eseguito successivamente all'interno della parola.

Altro : Altri parametri che nel test si trovano con il nome di: *Collisione tra le lettere, Misura incoerente tra lettere con e senza estensione in altezza, Irregolarità della grandezza delle lettere, Lettere atipiche, Traccia instabile, Scrittura incerta o tremolante ed infine Forme ambiguë delle lettere* non vengono trattati. Questo poichè alcuni sono racchiusi

parzialmente nelle parti sopra descritte, altri perché non si è trovato un modo efficiente per prelevare queste informazioni dalle sequenze di punti. Si lascia ad eventuali studi futuri un approfondimento anche in base a tecniche differenti di analisi della scrittura.

2.3 Creazione Features

Riprendendo le considerazioni della Sezione 1.3, le feature sono una delle componenti principali che influenzano la riuscita della categorizzazione. Infatti il nostro scopo è quello di estrarre delle feature dai dati a disposizione (fogli digitali), in modo da poter capire nella prima parte del progetto se uno scrittore sta usando la mano principale o la mano secondaria e nella fase successiva analizzare la scrittura dei bambini per distinguere un bambino DSA da uno non DSA.

Quello che faremo di seguito può essere schematizzato in tre parti principali:

Parte 1 : Questa parte riguarda estrarre le feature secondo le direttive del test BHK, e dunque quelle descritte in 2.2.4. Una volta ottenute queste feature sottoforma di vettore binario una prima analisi sarà quella di seguire lo standard del test, e dunque replicare una tabella simil-specie a quelle che il test prevede valutando il documento in base alla somma dei valori del vettore; una seconda analisi riguarda l'utilizzo dello stesso vettore binario, ma con l'ausilio di tecniche introdotte nel Capitolo 1, cioè usando delle Reti Neurali MLP.

Si noti che l'analisi in forma tabellare verrà fatta solo nel caso di dati del secondo dataset, quello riferito ai bambini delle scuole medie, poiché

l'analisi del test BHK si basa sul punteggio che il bambino ha conseguito rispetto al punteggio medio della classe scolastica di appartenenza.

Parte 2 : Questa parte, a seguito di alcuni risultati che commenteremo nel Capitolo 4, ha come obiettivo quello di estrarre delle feature meno elaborate differenziandosi dallo standard BHK. Le feature hanno la particolarità di non essere binarie (infatti le feature del BHK si presentano in maniera binaria: in una certa riga c'è o non c'è una particolare caratteristica). Ed anche in questo caso si procede con l'utilizzo di Reti Neurali (MLP) per l'analisi.

Parte 3 : Questa ultima parte esegue un tipo di analisi che si differenzia in modo sostanziale dalle prime due. Infatti mentre prima si è proceduto alla realizzazione di due Reti Neurali MLP per l'analisi delle feature estratte "a mano", in questa parte si procederà con l'utilizzo di una Rete Neurale CNN "ad hoc" addestrata a riconoscere lo scrittore a partire da immagini di righe in corsivo e vedremo se il comportamento di questa rete segue l'andamento delle precedenti e se migliora o meno il risultato del nostro problema.

Dunque procediamo con la creazione dei dataset, e l'estrazione delle feature, per analizzare poi nel Capitolo 4 gli esperimenti ed i loro risultati.

2.3.1 BHK Standard

Riferito a quanto detto nella Sezione 2.2, creiamo un vettore seguendo l'ordine delle 13 caratteristiche del test, dove le prime due caratteristiche riguardano la pagina completa, mentre le altre sono assegnate ad una riga. Aggiungiamo un'ulteriore campo al vettore della pagina, e quindi le caratteristiche per la pagina divetano 3. Dunque avremo un vettore di lunghezza

numero parole scritte	grandezza della scrittura	margine sinistro non allineato
int	[0 ; 5]	[3 ; 9]

Tabella 2.1: Caratteristiche della pagina

andamento altalenante	spazio insufficiente	angoli acuti	collegamenti interrotti
[0 ; 1]	[0 ; 1]	[0 ; 1]	[0 ; 1]

collisione	grandezza irregolare	misura incoerente	lettere atipiche
[0 ; 1]	[0 ; 1]	[0 ; 1]	[0 ; 1]

forme ambigue	lettere ritoccate	traccia instabile
[0 ; 1]	[0 ; 1]	[0 ; 1]

Tabella 2.2: Caratteristiche della riga

fissa, composto da: 3 elementi per la pagina (rappresentati nella Tabella 2.1) + $5 * 11$ elementi delle righe (gli 11 sono rappresentati nella Tabella 2.2).

Fatto questo tipo di selezione, si ottiene un dataset composto di 58 colonne, ed in base ai dati pre-selezionati otteniamo 102 righe per il primo dataset degli universitari e 91 righe per il secondo dataset dei bambini delle medie.

La prima tecnica che utilizziamo è quella inherente al vero test BHK, cioè eseguire la somma dei vettori binari per riga; ne parleremo più approfonditamente nella Sezione 4.1.

La seconda tecnica utilizza una Rete Neurale MLP per la classificazione. Viene usato il Framework Keras [29] con backend Tensorflow [30] che permettono di creare in maniera semplice una Rete Neurale MLP fully-connected. La nostra rete per questo tipo di esempi sarà formata di 58 nodi di input (riprendendo la terminologia della Sezione 1.2) nel primo livello, successivamente vi saranno due livelli nascosti di 40 nodi, fino a terminare nel livello di output composto da 2 soli nodi che rappresentano mano principale e mano

secondaria.

Per i livelli nascosti si usa come funzione di attivazione la funzione *Relu* mentre per l'ultimo livello viene utilizzata la funzione *SoftMax* per normalizzare l'output a somma unitaria e rappresentare dunque la probabilità di una classe piuttosto che un'altra. In questo esempio si addestra la rete in 200 epoche e con 10 righe come batch-size.

2.3.2 BHK modificato

Commenteremo in modo adeguato nel Capitolo 4 i risultati del test precedente nella Sezione 3.1, ma in linea di messima si può dedurre che una forma binaria delle caratteristiche potrebbe non essere il metodo migliore per estrarre le feature, e per rappresentarle. Inoltre può darsi che alcune fra le feature che abbiamo utilizzato si ripetano a livello di riga, facendoci intuire che probabilmente il testo scritto con la mano principale presenta delle caratteristiche comuni proprio fra i componenti del testo, piuttosto che in ogni singola parola di un altro foglio.

A seguito di queste considerazioni si analizzano delle caratteristiche del testo più "grezze" rispetto a quelle elaborate secondo lo standard del test BHK, delle quali vengono prelevati i dati così come sono, e dunque non esiste nessun range pre-impostato per il quale attribuire i valori ai campi del vettore.

Nelle Figure 2.13 e 2.14 possiamo notare a cosa corrispondono i valori che scegliamo per costruire il vettore di feature da dare alla rete MLP.

Vediamoli più nel dettaglio:

a1 : È la misura del margine sinistro, se si tratta di parole allora è lo spazio fra la parola precedente e la parola corrente. Se invece si tratta di righe, è la distanza fra l'inizio della riga e il margine sinistro del foglio.

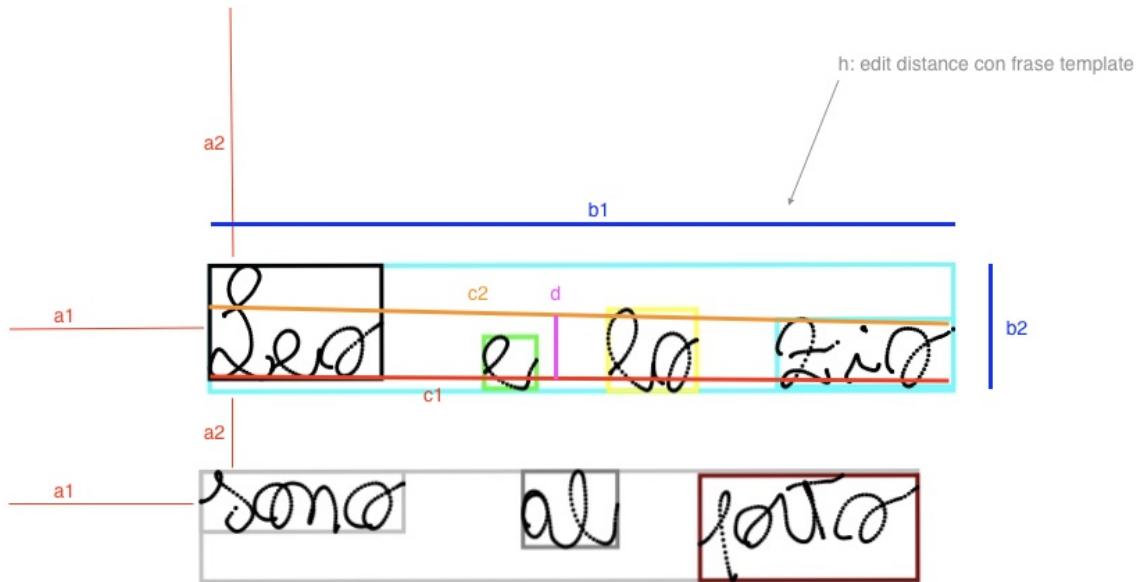


Figura 2.13: Feature "grezze" prelevate fra le righe con BHK modificato

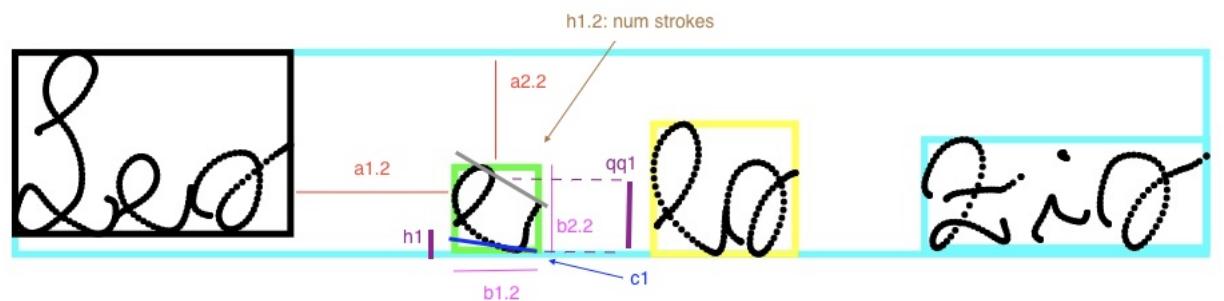


Figura 2.14: Feature "grezze" prelevate fra le parole con BHK modificato

a2 : Questo parametro in una riga indica la distanza fra il margine superiore del rettangolo attorno alla riga e l'elemento superiore più vicino. Infatti se si tratta della prima riga sarà la distanza fra la linea superiore della riga ed il margine superiore del foglio, mentre se si tratta di una roga intermedia diventa la distanza fra le righe. Notiamo che questa può essere anche negativa, nel caso in cui due righe si intersecano. Se si riferisce alle parole all'interno della riga allora indica lo spazio bianco che vi è fra il margine superiore della parola e il margine superiore della riga che comprende tale parola.

b1 : Nelle righe indica la lunghezza complessiva della riga, nelle parole invece la lunghezza della parola. Questi parametri si presentano in pixel come unità di misura.

b2 : Nelle righe e nelle parole indica l'altezza rispettivamente della riga o della parola.

c1 : Questo parametro rappresenta la pendenza della retta di migliore approssimazione che approssima i minimi locali. Nel caso di riga i minimi locali riguardano tutti i minimi delle parole; nel caso delle parole la retta dei minimi approssima solo i minimi della parola. Questa pendenza è il coefficiente angolare della retta.

c2 : Questo parametro rappresenta la pendenza della retta di migliore approssimazione che approssima i massimi locali. Nel caso di riga i massimi locali riguardano tutti i massimi delle parole; nel caso delle parole la retta dei massimi approssima solo i massimi della parola. Questa pendenza è il coefficiente angolare della retta.

d - qq : È un parametro composto dal coefficiente q_1 della retta dei minimi, e il coefficiente q_2 della retta dei massimi. Indica la distanza che c'è fra la retta superiore e quella inferiore.

h : Nelle righe indica l>Edit Distance fra il testo del template della riga ed il testo compreso nel txt di Inkspace. Nelle parole indica l>Edit Distance fra il template della singola parola e il testo compreso nel txt.

n : È un parametro che indica il numero di stroke che sono presenti nelle parole. Questo campo è presente solo nelle feature che appartengono alle singole parole.

In base a come sono scritti gli indici di queste caratteristiche, esse si riferiscono ad una riga o ad una parola all'interno di una riga. Per esempio riguardo ad una specifica riga 8 di queste caratteristiche:

$a_1, a_2, b_1, b_2, c_1, c_2, d, h$: che assumono i significati precedentemente descritti, ovviamente riferiti alle righe;

$a_{1.1}, b_{1.1}, b_{2.1}, c_{2.1}, n_{.1}, h_{.1}, qq_{.1}$: che assumono i significati precedentemente descritti, ovviamente riferiti alle righe, e si comprende anche di quale parola si stia parlando all'interno della riga poiché a seguito del a_1 vi è un punto con l'indice della parola nella riga (es. $a_{1.3}$ indica il parametro a_1 riferito alla 4° parola);

Questa nuova definizione delle strok viene fatta in due modi diversi che descriveremo meglio nel Capitolo 3, ma si tratta di selezionare all'interno di ogni riga tutte le parole, oppure solo le due più importanti. Le due più importanti sono decise in base alla grandezza della parola, e cercando di non prendere informazioni riguardo alle "stop-words".

2.3.3 Metodo Offline

Il terzo metodo che utilizziamo è un metodo che va a perdere le informazioni che tanto abbiamo faticato ad ottenere: le informazioni online. Questo metodo utilizza le reti convoluzionali CNN addestrate sulla scrittura in corsivo di 50 scrittori statunitensi utilizzando il dataset IAM [31]. Il progetto parte da un prototipo online [32] [33], del quale si mantiene la rete CNN ed il metodo di creazione delle "patch".

La rete viene dunque addestrata sulla scrittura ricevendo in ingresso righe in corsivo scritte a mano, da queste apprende le feature della scrittura in corsivo, che utilizziamo per fare il pre-processing alle immagini del nostro dataset chiamato fino ad adesso "fittizio". In base alle feature che ha appreso, per questo sarebbe una buona pratica fare apprendimento su IAM per più volte con seed randomici diversi, si avranno in uscita agli stadi convoluzionali (e più precisamente dopo il flattening) un insieme di nodi che diventano il livello di input per una rete fully-connected. Tutto quello che sta dopo (cioè la rete MLP) deve essere appresa, e questo apprendimento viene fatto sul dataset che abbiamo creato.

Durante il secondo apprendimento possiamo assumere due comportamenti differenti rispetto ai pesi pre-learned della rete CNN: possono essere modificati o non possono essere modificati. La tecnica di non rendere possibile la modifica di tali pesi con il back-propagation si chiama "freezing" ed i pesi si dicono "freezed". Infatti potremmo non essere interessati a speciaizzare la rete CNN sul nostro dataset, ma mantenerla generica per la scrittura in corsivo.

Anche il dataset, e la sua costruzione, possono essere caratterizzati da due comportamenti: mantenere i dati evidenziando la cronologicità dei punti, e la velocità nel trascriverli, oppure perdere questo tipo di informazione. Questo

scenario ci porta alla creazione di due dataset distinti: uno in cui ogni tratto è caratterizzato da un insieme di punti disgiunti, di tipo circle in SVG, e quindi elementi separati anche nella definizione dell'elemento all'interno del documento; un altro in cui i tratti sono oggetti di tipo polyline, e quindi i punti singli non sono ben visibili, e si perde informazione sulla velocità con la quale sono stati tracciati tali punti, anche se si la rappresentazione finale è più realistica e simile a quella realmente tracciata.

Capitolo 3

Dataset

Come specificato nel Capitolo 1, non si dispone di un dataset, ed è necessario crearne uno. In realtà durante il progetto ne sono stati creati ben due: uno prodotto da studenti universitari usando la mano principale e la mano secondaria che simula una forte disgrafia; ed un altro prodotto da bambini della scuola media statale di Scarperia e San Piero dei quali si annotano caratteristiche DSA o meno.

Il primo dataset prodotto dagli studenti universitari presenta delle forti differenze, anche visibili, fra i documenti scritti con la mano principale e quelli scritti con la mano secondaria. Come vedremo nel capitolo dei risultati, infatti, l'accuratezza con la quale distingue le due classi è relativamente alta.

Un esempio di dati di questo primo dataset è dato dalle immagini in Figura 3.1.

3382188

Leo e lo zio
 sono al porto
 magari un gelato
 con loro li sanno
 Nia e Rine

3382188

Leo e lo zio
 sono al porto
 magari un gelato
 con loro li sanno
 Nia e Rine

(a) Test eseguito con la mano principale. (b) Test eseguito con la mano secondaria.

Figura 3.1: Esempio di documenti prodotti da studenti universitari.

Il secondo dataset è stato prodotto da bambini di 11 anni di età, tra i quali vi era una bassa percentuale di DSA (10%). Inoltre non tutti questi bambini avevano una scrittura considerevolmente diversa dai coetanei, ma piuttosto andava ad uniformarsi a quella degli altri bambini. Ed in secondo luogo, vi erano alcuni fra i bambini non DSA che avevano un tratto più incerto rispetto agli altri. Vedremo nel Capitolo 4 cosa questo ha comportato.

Un esempio di dati di questo secondo dataset è dato dalle immagini in Figura 3.3 e la Figura 3.2 riporta uno dei bambini nell'atto dell'esecuzione del test.

Per ognuno di questi due gruppi di dati si sono creati i tre tipi diversi di dataset che tratteremo in questo capitolo.

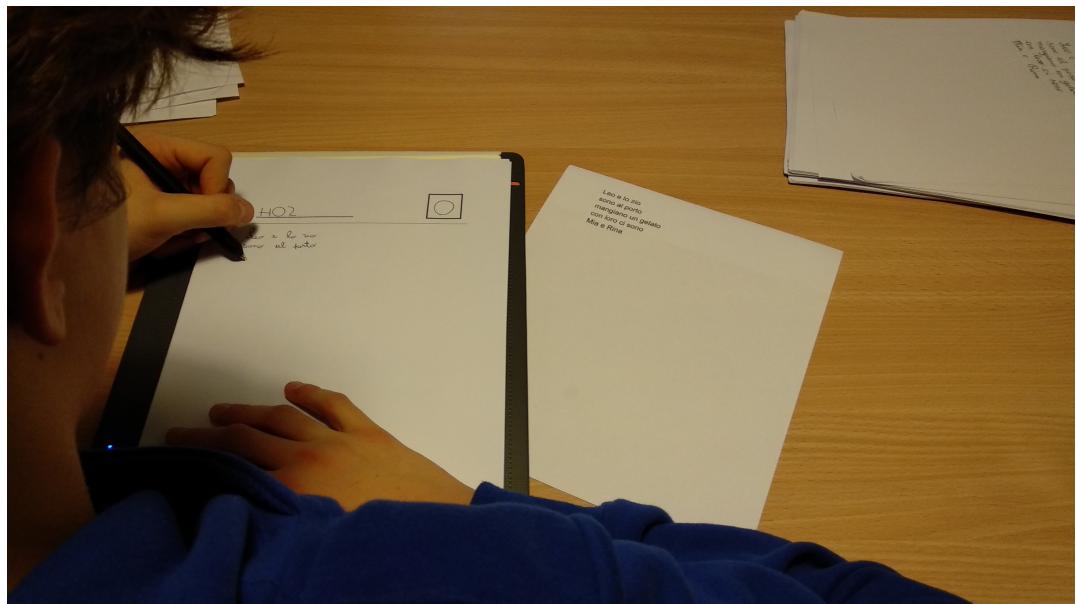


Figura 3.2: Bambino di 11 anni mentre esegue il test sottoposto

BO 7

Leo e lo zio
sono al porto
mangiamo un gelato
con loro ci sono
Mia e Rina

BO 9

Leo e lo zio
sono al porto ~~mangiamo~~
Mangiamo un gelato
con loro ci sono
Mia zia Mina

(a) Test eseguito da un bambino non DSA. (b) Test eseguito da un bambino DSA.

Figura 3.3: Esempio di documenti prodotti da bambini di prima media.

3.1 Dataset BHK standard

Abbiamo discusso nella Sessione 2.3.1 delle features che analiziamo con questo metodo. Otteniamo così un dataset composto di 58 colonne, ed un totale di 102 righe rappresentanti ognuna un foglio prodotto da una mano (principale o secondaria) di uno scrittore. Ovviamente alle 58 colonne è necessario aggiungere una colonna di informazione sulle label attribuite ad ogni riga.

3.2 Dataset BHK modificato

In questo caso creiamo, come descritto nella Sessione 2.3.2, due dataset differenti. Un primo sarà composto da 159 colonne, caratterizzate da 5 righe che hanno un numero di feature pari a 8 per riga; ogni riga con un numero variabile di parole, ed ad ogni parola sarà associato un numero fisso pari a 7 features. Dunque come precedentemente accennato:

$$(5\text{righe} * 8\text{feature}) + (7\text{features} * 4\text{parole}) + (7\text{features} * 3\text{parole}) + \\ (7\text{features} * 3\text{parole}) + (7\text{features} * 4\text{parole}) + (7\text{features} * 3\text{parole})$$

poichè le righe sono composte di 4, 3, 3, 4, 3 parole rispettivamente.

Se si osservano le parole che ci sono in ciascuna riga, capita di imbattersi in quelle che in letteratura sono chiamate "stop-word", cioè le congiunzioni, gli articoli, le preposizioni, ecc... . Questi tipi di parole portano con se dell'informazione utile, poichè ogni dettaglio è importante in casi come questo dove non si conosce in alcun modo quale sia l'entità che possa distinguere la mano principale dalla secondaria, pittosto che un DSA da un non DSA. Si sperimenta, ad ogni modo, la costruzione del dataset evitando le "stop-word", e quindi selezionando per ogni riga solamente due parole, la prima e l'ultima. E quindi la dimensione del vettore delle features diminuisce diventando:

$(5\text{righe} * 8\text{feature}) + (7\text{features} * 2\text{parole}) + (7\text{features} * 2\text{parole}) + (7\text{features} * 2\text{parole}) + (7\text{features} * 2\text{parole}) + (7\text{features} * 2\text{parole})$

che in totale fanno 110, e con la colonna delle label: 111 features pero ogni riga. Entrambi i dataset hanno lo stesso numero di occorrenze: 94.

3.3 Dataset CNN

Il terzo e ultimo metodo di estrarre le features che abbiamo preso in considerazione riguarda una suddivisione dei dati in modo da prepararli per una rete convoluzionale. Le reti convoluzionali lavorano con le immagini, quindi il primo pre-processing che viene eseguito è quello di trasformare i documenti SVG in immagini. Questo viene fatto pensando a che tipo di rete va costruita in seguito ed in base ai materiali che disponiamo.

Abbiamo a disposizione, come descritto nella Sessione 2.3.3, un dataset di righe in corsivo, scritte a mano, chiamato IAM, prodotto da 250 scrittori che hanno scritto oltre 115k parole in lingua inglese. Utilizziamo questo dataset per allenare la rete CNN a riconoscere gli scrittori in modo da apprendere le feature che riguardano la scrittura in corsivo.

A questo punto pensiamo a come strutturare il nostro dataset, nonchè a come frammentare i vari fogli scritti, già preprocessati, e quindi dei quali è possibile separare le righe e successivamente le parole all'intero delle righe. Utilizziamo la stessa struttura del dataset IAM cioè separando in cartelle per scrittore, senza perdere informazione su quale scrittore ha eseguito il testo. Viene creata una cartella per ogni scrittore, si capisce lo scrittore dalla prima riga del testo scritto che, a seguito del pre-processamento, viene eliminata. Viene compresa, sempre dalla prima riga, anche la classe di appartenenza, cioè "O" se mano principale, e "X" se mano secondaria per i dati degli stu-

denti universitari, ed "O" non DSA e "X" DSA per i dati prelevati ai bambini di scuola media. A questo punto si hanno le cinque righe successive, ben separate e delle quali conosciamo lo scrittore, e la classe di appartenenza. Si prosegue creando un file SVG contenente solo la riga necessaria, e convertendo tale file SVG in un file immagine PNG rinominandolo opportunamente con il nome univoco del file WILL da cui è stato estratto, seguito dalla classe di appartenenza ("O" oppure "X") ed evidenziando per ogni immagine anche il numero di riga corrispondente ("row_0" se si tratta della prima riga).

A questo punto è necessario fare un ulteriore considerazione: nella trasformazione in 2.1, e nelle successive elaborazioni, si ottengono un insieme di punti che rappresentano, uniti fra di loro, gli stroke. Per lo studio ed il successivo pre-processing ci siamo trovati a trattare le coordinate dalle quali far passare il tratto come se fossero puntiformi.

Questo ha portato a notare due fatti:

- se si rappresenta lo stroke come polyline (cosa che viene fatta inizialmente) si mantiene l'informazione precisa del tratto, come lo si vede sul foglio dove viene fatto il percorso con la penna, ma si perde informazione sulla velocità con la quale il tratto è stato tracciato;
- se si rappresentano come insieme di punti separati si mantiene l'informazione della velocità, poiché nei tratti meno salienti dove la penna scorre più veloce ci saranno meno punti, mentre nei tratti più salienti il numero di punti addensati sarà maggiore.

Quindi la decisione è stata quella di creare due dataset, composti di immagini PNG delle righe, costruite a partire sia dagli SVG con polyline (cioè con le linee reali), sia con i punti separati (cioè con circonferenze di piccolo diametro

centrate nei punti dell'SVG). Discuteremo nel Capitolo 5 un possibile utilizzo futuro di questo secondo dataset di "punti".

Capitolo 4

Esperimenti e Risultati

In questo capitolo si discutono gli esperimenti che sono stati svolti, si danno le motivazioni sul perchè sono stati costruiti ben 3 dataset differenti (esposti nel Capitolo 3), quali tecniche sono state utilizzate per ognuno dei differenti dataset e si introduce l'ultima parte del progetto: l'analisi della scrittura dei bambini delle scuole Medie di Scarperia e San Piero.

Vediamo più approfonditamente gli esperimenti svolti in questo progetto di tesi.

4.1 Esperimenti BHK standard

Gli esperimenti che vengono svolti in questa sezione sono due:

esperimento 1 : consiste nel sommare i vettori binari, e calcolare per ogni gruppo di dati la media e la deviazione standard;

esperimento 2 : riguarda l'utilizzo sui vettori binari di una Rete Neurale MLP per distinguere i dati nelle due classi di appartenenza (mano principale e mano secondaria per il primo dataset, e DSA o non DSA per il secondo).

I due dataset verrano però trattati in maniera differente.

4.1.1 BHK standard sul dataset degli universitari

Su di questo dataset verrà svolto solo il secondo esperimento, poiché come specificato in 2.3, non ha senso applicare il primo esperimento sul dataset degli universitari.

Esperimento 2

Viene costruito un dataset secondo il modello BHK standard (Sezione 3.1) a partire dai documenti degli universitari. Si ottengono 102 occorrenze binarie. A seguito dell'utilizzo di questi vettori in una Rete Neurale MLP fully-connected si sono ottenuti risultati molto vari e poco accurati, poichè nelle 102 occorrenze si presentano ripetizioni di valori, anche con label differenti. Infatti questo dataset non è fatto per essere utilizzato con una rete poichè in questo esperimento i risultati variano in base alla divisione del dataset in train, test e validation. Inoltre al variare del random seed cambia completamente la riuscita o meno di una corretta categorizzazione, con picchi di accuratezza elevati e immediate ricadute. Abbiamo provato ad ovviare a questo problema mediando accuratezza e loss sui risultati di più iterazioni, tramite una suddivisione con k-fold cross validation, anche se nella Tabella 4.1 possiamo notare un'accuratezza molto bassa, indice di non riuscita della tecnica usata.

Tabella 4.1: Primo Dataset

n° iter	1	2	3	4	5	6	7	TOT.
Loss	2.65	0.25	0.95	2.09	2.81	0.55	1.84	1.92 (± 1.41)
Acc	0.45	0.90	0.72	0.5	0.5	0.8	0.6	0.65 (± 0.19)

Risultati training del primo dataset con K-fold cross validation

Il corretto modo di utilizzare questo dataset di vettori binari potrebbe essere secondo il Test BHK, cioè sommando le righe di ogni tupla, calcolando il valore per ogni pagina di ogni soggetto in una classe, e calcolando la media e la varianza. Vediamo come si comporta il dataset dei bambini delle scuole medie, questa volta con entrambi gli esperimenti 1 e 2 in 4.1.

4.1.2 BHK standard sul dataset dei bambini

Il dataset prodotto dai bambini può essere analizzato secondo entrambi i metodi. Vediamo l'esperimento 1 come se la cava sui dati dei bambini.

Esperimento 1

In generale viene applicata una fase preliminare di pre-processing sui dati prelevati, che sono divisi per classi scolastiche di appartenenza. Le informazioni di nostro interesse sono situate nei vettori binari del dataset ed eseguiamo su questi la somma dei valori per ogni riga che corrispondente ad un documento prodotto da un bambino. Si annotano i valori di somma, etichettandoli con la classe del documento. Il test BHK suggerisce di calcolare per ogni sezione la media e la varianza delle somme dei valori delle feature. Un documento è considerato prodotto da un bambino DSA quando la somma delle sue feature sta al di sopra di una certa soglia. La Tabella

4.2 mostra i valori di media e deviazione standard per ogni classe scolastica, dove consideriamo le classi come le sezioni in cui i bambini sono stati inseriti nella scuola media.

Potrebbe però non avere molto senso calcolare i valori di media e deviazione standard per ognuna delle classi scolastiche, poiché le classi che siamo analizzando in questo esperimento corrispondono a DSA e non DSA, classi che sono poco legate alla classe di appartenenza dei bambini (per quanto riguarda le scuole medie).

Si prosegue alla suddivisione del dataset in due classi sole: DSA e non DSA; ed in base a questa suddivisione abbiamo proceduto calcolando per ogni classe gli stessi parametri di media e deviazione standard. La Tabella 4.3 mostra questi valori, e la Figura 4.1 mostra il rapporto fra la Gaussiana della classe non DSA e quella della classe DSA.

Non DSA	punteggio minimo	10	DSA	punteggio minimo	12
	punteggio massimo	23		punteggio massimo	23
	media	15.61		media	16.11
	dev.std.	3.15		dev.std.	3.03
n° alunni: 82		n° alunni: 9			

Tabella 4.3: Analisi dataset scuola media

Media e varianza dei documenti fra gruppo non DSA e gruppo DSA.

n° alunni: 22	classe A	punteggio minimo	11	n° alunni: 20	classe B	punteggio minimo	10
		punteggio massimo	23			punteggio massimo	19
		media	16.77			media	14.05
		dev.std.	3.13			dev.std.	2.42
n° alunni: 22	classe C	punteggio minimo	10	n° alunni: 11	classe E	punteggio minimo	11
		punteggio massimo	23			punteggio massimo	22
		medie	15.41			media	16.09
		dev.std.	3.30			dev.std.	3.65
n° alunni: 11	classe F	punteggio minimo	13				
		punteggio massimo	20				
		medie	16.55				
		dev.std.	2.23				

Tabella 4.2: Analisi dataset scuola media

Media e varianza dei documenti in ogni classe.

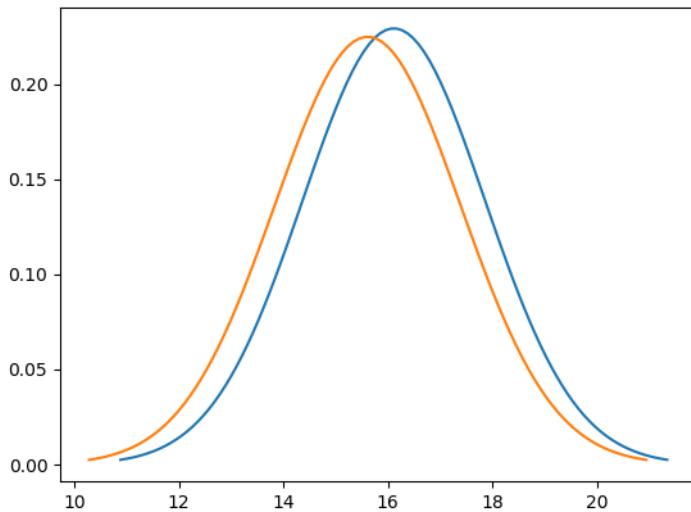


Figura 4.1: Gaussiane dei valori bambini non DSA (arancione) e DSA (blu)

La regola del test BHK, dopo aver calcolato la somma dei parametri del vettore per ogni documento, sarebbe quella di considerare DSA i bambini che hanno prodotto un documento che supera di due punti percentuali la media della classe. In questo caso possiamo notare come non vi sia una forte differenza fra la gaussiana dei DSA e quella dei non DSA (possiamo vederlo dalla somiglianza delle due funzioni nella Figura 4.1) ed infatti le righe binarie, essendo molto imprecise, si somigliano fra di loro, anche se appartenenti a classi diverse.

Purtroppo il test non ha avuto riscontri positivi, poiché risulta molto erato e con valori di precisione molto bassi. Si lascia a degli studi futuri una implementazione migliore della fase di pre-processing ed estrazione delle feature, in modo da ottenere valori più alti delle metriche utilizzate (accuratezza, precision e recall).

4.2 Esperimenti BHK modificato

In questo secondo esperimento andremo a costruire due dataset per ognuno dei precedenti dataset (universitari e medie). Costriremo i dataset basandoci su quanto specificato nella Sezione 3.2, e dunque collezionando più informazioni rispetto a quello che generava vettori binari nell'esperimento nella precedente Sezione 4.1).

Vengono create due reti neurali MLP con due soli livelli nascosti di 40 neuroni ciascuno, usando la funzione di attivazione relu, e senza l'utilizzo di livelli di dropout (azione che elimina temporalmente delle connessioni fra neuroni, utilizzata per irrobustire la rete). La prima rete avrà nel primo livello di input 158 neuroni, la seconda 110, in base al numero di parole per riga che vengono considerate. Entrambe le reti hanno un livello di output composto da 2 neuroni con funzione di attivazione softmax, che normalizza gli output con la somma totale degli output, ed attribuisce ad ogni neurone una probabilità. Quindi l'input appartiene alla classe con probabilità maggiore.

4.2.1 BHK modificato sul dataset degli universitari

Poichè il dataset è bilanciato (infatti ogni universitario ha scritto sia con la destra che con la sinistra), si possono notare due comportamenti principali: il primo riguarda il fatto che durante le fasi di apprendimento i valori di accuratezza e di loss variano molto, il secondo invece riguarda l'importanza che hanno dei parametri come la precision e la recall su una buona comprensione di ciò che sta accadendo. Infatti vi sono dei casi in cui con la destra e con la sinistra uno scrittore scrive con lo stesso spazio a sinistra, o lasciando un buon margine fra le parole, e non tenendo conto di parametri come la velocità o il tremolio della mano nel tracciare il gesto si ha che la

mano principale può essere scambiata per la mano secondaria. Viene scelta la strada di utilizzare una k-fold cross validation per poter fare un confronto fra i valori di recall e precision mediati su tutte le k iterazioni ed evitare in questo modo di ottenere valori di accuratezza e loss non dei migliori (rispettivamente bassa accuratezza e alta la loss). Si nota con questi esperimenti che le due classi hanno una recall, in base al numero degli elementi della classe, accettabile (con molta tolleranza). Vediamo nelle Figure 4.2 e 4.3 le matrici di confusione mediate sulle k iterazioni e nelle Tabelle 4.4 e 4.5 i valori di accuratezza, anch'essi mediati su k . Nonostante siano degli stessi esperimenti, vengono rappresentate due Matrici di confusione e due tabelle con valori di accuratezza poichè, fra le tante esecuzioni degli algoritmi di MLP, queste rappresentano i valori medi che si sono ottenuti variando i parametri di learning rate ed il numero di epoche.

Tabella 4.4: Media delle accuratezze sul test (100 epoche) con deviazione standard, dataset universitari

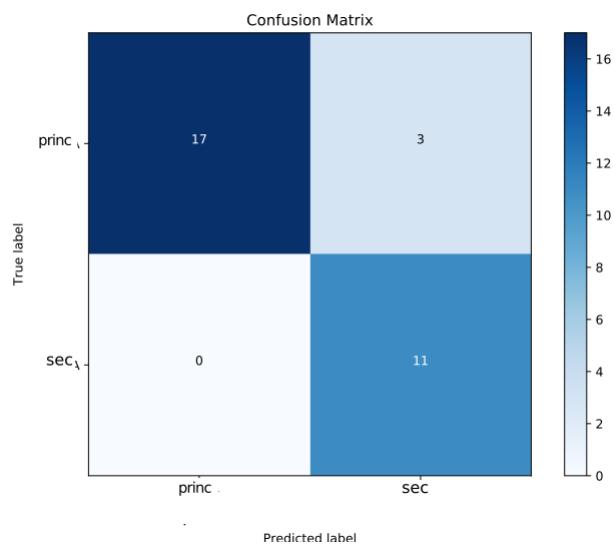


Figura 4.2: Matrice di confusione mediata su k , dataset universitari (100 epoche)

ACCURATEZZA	
70.97%	(+/-9.60%)

Tabella 4.5: Media delle accuratezze sul test (300 epoche) con deviazione standard, dataset universitari

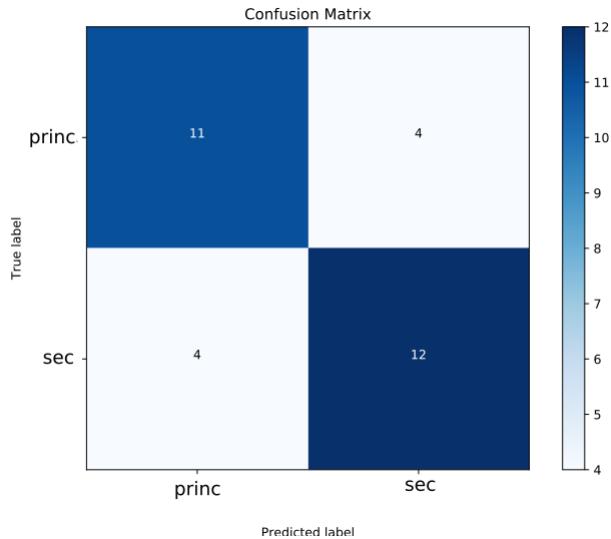


Figura 4.3: Matrice di confusione mediata su k , dataset universitari (300 epoche)

Si può notare che la recall è sempre relativamente alta, e che in media si ottengono dei valori di accuratezza del 70%. Un bel risultato rispetto al metodo di utilizzo dei dati del caso precedente nella Sezione 4.1.

4.2.2 BHK modificato sul dataset dei bambini

Anche in questo caso, nonostante i risultati siano più ordinati del tentativo sul precedente dataset, si ha che l'accuratezza (così come la loss) variano molto. Si effettua quindi una k-fold cross validation per calcolare la dispersione dei valori di loss e accuratezza, che sembrano poterci soddisfare, se non si fosse calcolato anche le metriche di Recall e Precision dalla matrice di dispersione. Vengono rappresentate in Figura 4.4 e 4.5 le matrici di confusione medie relative a due impostazioni differenti del learning rate; nelle rispettive Tabelle 4.6 e 4.7 si indicano i valori medi di accuratezza del problema.

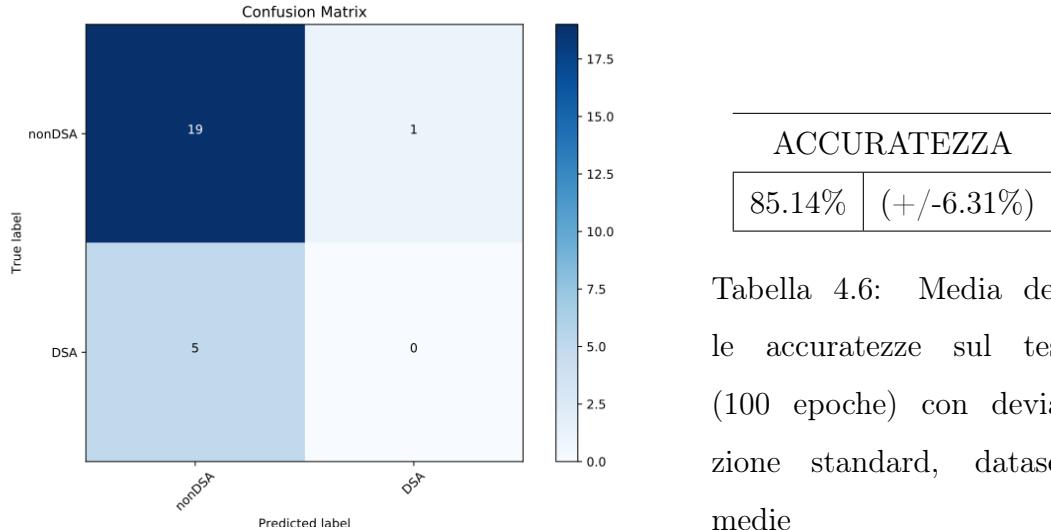


Tabella 4.6: Media delle accuratezze sul test (100 epoche) con deviazione standard, dataset medie

Figura 4.4: Matrice di confusione mediata su k , dataset medie (100 epoche)

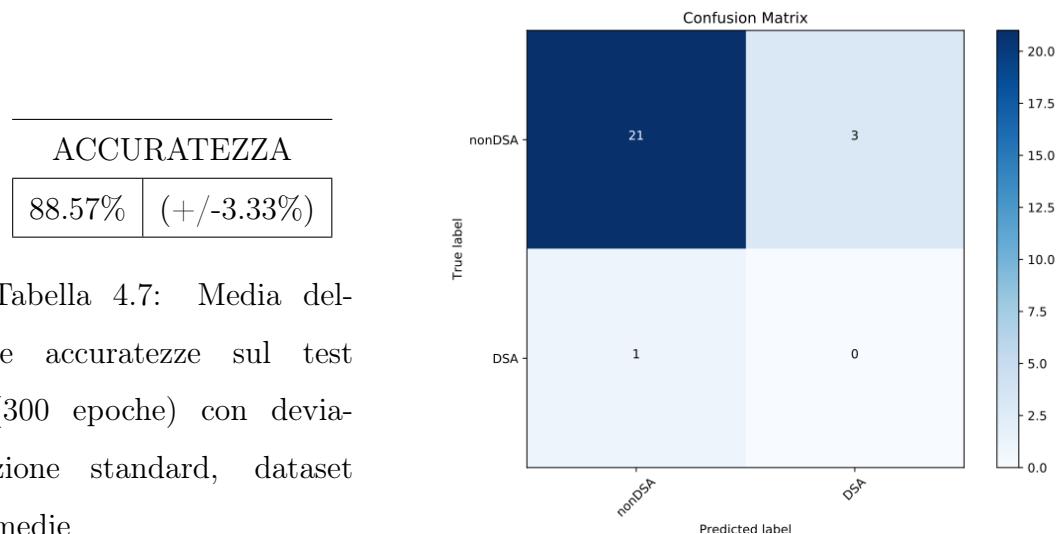


Tabella 4.7: Media delle accuratezze sul test (300 epoche) con deviazione standard, dataset medie

Figura 4.5: Matrice di confusione mediata su k , dataset medie (300 epoche)

Anche per il secondo dataset, quello contenete per ogni riga solo la prima e l'ultima parola, viene usata una rete neurale MLP con diverso numero di neuroni nel primo livello di input (110 neuroni invece di 158). Ed anche per questo esperimento viene usata una k-fold cross validation con un seed pseudo-randomico (generato dalla funzione `randint()` di `sklearn`). Si evita di inserire le rispettive Tabelle e Figure dato che l'accuratezza, nonostante alta, non conta molto in questo problema; vengono valutate altre metriche.

Si può quindi notare come questo problema sia mal bilanciato, con uno sbilanciamento indicativamente di 1 : 8.25 fra il numero di dati non DSA e quelli DSA. Nonostante si sia aggiunto alle reti la metrica di precisione e di recall, in modo da non modificare i pesi solo osservando l'accuratezza, i risultati che otteniamo sono ben lontani da un possibile utilizzo pratico delle tecniche viste sino ad adesso. Su un insieme come questo le uniche soluzioni possono essere quello di valutare anche delle features riguardanti il tempo di percorrenza che la penna impiega a tracciare il segno, ed altre caratteristiche di movimenti rapidi, angoli acuti, simboli non propriamente comprensibili.

4.3 Esperimenti CNN

Nonostante si noti un miglioramento rispetto alla prima tecnica, non si è ancora del tutto soddisfatti. Si è voluto apporfordire una tecnica offline di estrazione delle features poichè i dati a nostra disposizione sono ancora pochi e tecniche di creazione sintetica dei dati (come il data augmentation) vanno a variare la struttura dei dati e a falsificare qualsiasi risultato. La tecnica offline che abbiamo deciso di esplorare è quella delle reti neurali convoluzionali.

A seguito della creazione del dataset descritto nella Sezione 3.3, viene implementata la rete neurale CNN a partire da un progetto riguardante il

riconoscimento dello scrittore fra 50 scrittori differenti, che hanno prodotto righe di testo in inglese. Il progetto che utilizziamo utilizza una rete CNN (rappresentata in Figura 4.6 e 4.7) composta di tre livelli convoluzionali 2D che mantengono le dimensioni dell’immagine originale con uno zero-padding, seguiti ciascuno da una funzione di attivazione relu e successivamente da una funzione di maxpooling per diminuire la dimensione delle immagini. A seguito dei tre livelli vi è una MLP composta di due livelli nascosti, collegata con un Flattening delle immagini. I due livelli nascosti sono composti rispettivamente di 512 e 256 neuroni, sempre con a seguire funzione di attivazione relu e applicazione di dropout. L’ultimo livello, quello di output, è composto da 50 neuroni che rappresentano ciascuno uno scrittore differente.

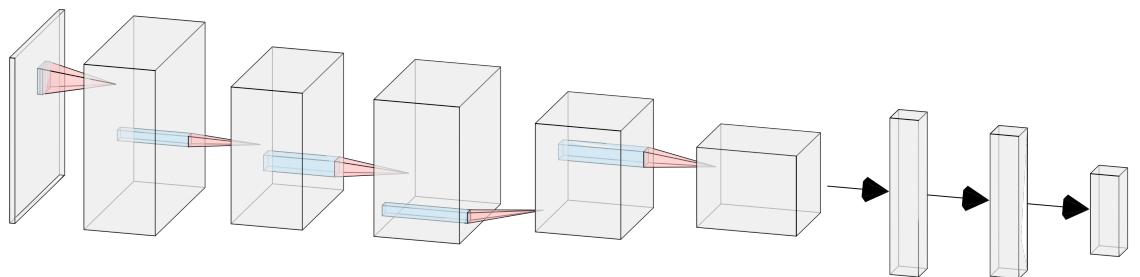


Figura 4.6: Rappresentazione AlexNet della CNN utilizzata prima della modifica (50 nodi di output invece di 2)[34]

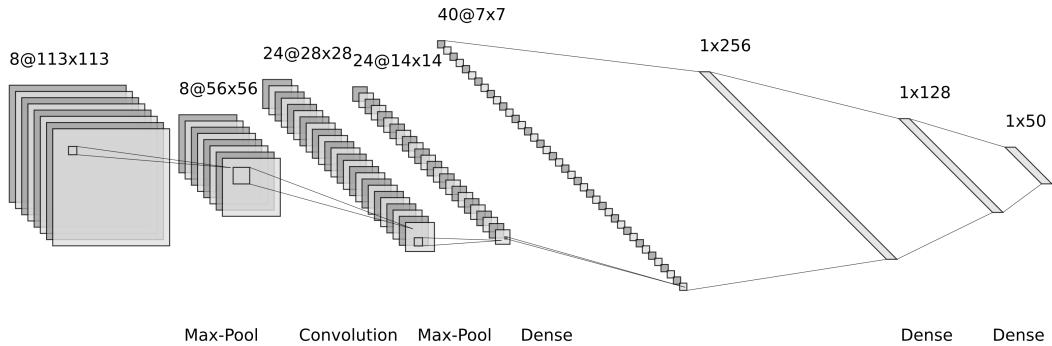


Figura 4.7: Rappresentazione LetNet della CNN utilizzata prima della modifica (50 nodi di output invece di 2)[34]

Questa rete così creata ci serve come punto di partenza per poter distinguere in primis la scrittura con la mano principale e secondaria degli studenti universitari, e in secondo luogo per distinguere la scrittura prodotta da un bambino DSA da quella di un bambino non DSA.

La rete prende in ingresso delle immagini di dimensione fissa, quindi è necessario un pre-processing sui documenti per prelevare immagini. Viene sfruttato il lavoro di separazione degli stroke in parole e successivamente in righe, che avevamo utilizzato nel Capitolo 2, e per ogni documento vengono prelevate 5 righe contenenti il testo del template. Ogni riga, che attualmente è in formato SVG, viene convertita in immagine png, e costruito il dataset di immagini PNG per ogni documento. Viene partizionato il dataset in due maniere differenti in base al metodo utilizzato, cioè in maniera randomica ogni riga indipendente, oppure considerando i gruppi di appartenenza delle immagini delle righe, cioè non dividendo mai le righe di un documento di uno

scrittore, ne quelle prodotte con la destra ne quelle prodotte con la sinistra. Quindi nel caso degli scrittori universitari, in questo secondo metodo, avremo righe della mano destra e sinistra o tutte nell'insieme di train, o tutte nell'insieme di test o validation. Nel caso del dataset dei bambini delle medie si mantiene integrità a livello di documento, quindi le 5 righe che compongono il documento devono restare insieme, ma non si hanno vincoli su documenti di bambini relativi alla stessa sezione di appartenenza.

Una volta separato il dataset in train, validation e test si prosegue a generare le patch di dimensione fissa da dare "in pasto" alla rete. Queste vengono generate in numero randomico, indicativamente per ogni riga vengono prelevate 300 immagini di dimensione fissa e date come ingresso alla rete. Da qui in poi il procedimento di aggiornamento dei pesi della rete (filtri convoluzionali per CNN e pesi per MLP) segue l'algoritmo di backpropagation introdotto nel Capitolo 1.

Vediamo come si distinguono gli approcci ai test da eseguire in base al dataset e alla composizione di questi.

4.3.1 CNN sul dataset degli universitari

Per distinguere i documenti prodotti dalla mano principale da quelli prodotti dalla mano secondaria negli studenti è stata inizialmente modificata la rete di partenza, togliendo l'ultimo livello fully-connected da 50 nodi, ed aggiungendone uno da 2 soli nodi che rappresentano rispettivamente la principale e la secondaria. Da questa struttura della rete CNN possiamo scegliere due strade per poter allenare la rete sul nostro dataset:

- La prima ipotesi è quella di partire "da 0" con uno stato randomico e generare i pesi iniziali, apprendendo da questi i pesi per la CNN che

estraggono le features importanti per poi passarle alla fully-connected che esce in 2 neuroni di output;

- La seconda ipotesi invece consiste nel partire dalla rete precedentemente addestrata (quella con il dataset IAM) e caricare i pesi che classificano meglio le scritture dei 50 scrittori. Cambiare l'ultimo livello inizializzando random i pesi dell'ultimo collegamento. A questo punto abbiamo altre due scelte:
 - aggiornare solo i pesi della rete fully-connected (che computazionalmente pesa di meno);
 - aggiornare anche i pesi di qualche strato convoluzionale, il che significa modificare i pesi che compongono i filtri di convoluzione;

In questo progetto come esperimenti finali si è deciso di implementare entrambe le sottocasistiche della seconda ipotesi: partire in entrambi i casi dalla rete CNN già appresa sui dati dei 50 scrittori in 3 modi diversi (che si differenziano in base al batch_size, al numero di epoche utilizzate e al learning rate) ed a seguito del caricamento di tali pesi sulla rete sperimentare l'aggiornamento dei pesi sul nostro dataset limitandoci ad aggiornare la fully-connected, o anche il primo livello convoluzionale (primo considerando il modello come uno stack, quindi l'ultimo strato convoluzionale aggiunto).

Si riporta la Tabella 4.8 dove vengono rappresentati i valori di loss e accuratezza sul validation (seguiti dai valori di accuratezza sul test) dividendo train, test e validation come descritto in 4.3.1. Infatti è stato logico abbandonare la strada di dividere randomicamente le righe negli insiemi, poichè la rete è fatta per estrarre le features dalle righe di testo scritte a mano e riconoscere lo scrittore fra altri 50; ed un analisi più accurata ci ha portato a pensare che se la rete ha già visto delle righe di un documento, diventa

un problema più semplice quello di riconoscere l'etichetta binaria di una riga simile ad alcune già viste, dato che questo si riconduce al problema per cui la rete è stata creata su IAM (ed ottiene tra l'altro un accuratezza sempre superiore al 85% dopo 8 epochhe, arrivando anche a 94% dopo 14 epochhe).

sottometodo	1			2		
learnig_rate	10^{-3}			10^{-4}		
batch_size	15			15		
epochhe	100			300		
pre-learned 1	34,03%	86,17%	84,18%	43,81%	82,92%	82,94%
pre-learned 2	29,90%	87,71%	87,33%	42,20%	83,05%	82,17%
pre-learned 3	42,01%	86,73%	83,77%	41,82%	84,50%	79,82%
loss	acc	ACC.	TEST	loss	acc	ACC.
validation	validation			validation	validation	TEST

Tabella 4.8: Accuratezza Validation rete CNN sugli universitari, sottometodo 1 vs. 2

Si nota come il problema di distinguere la mano principale dalla mano secondaria sia, non risolto, ma comunque ad un buon punto ottenendo anche picchi dell' 87% di accuratezza sul test.

4.3.2 CNN sul dataset dei bambini

Il problema di distinguere i bambini DSA dai bambini non DSA invece ha un riscontro non molto positivo con questa tecnica, infatti a seguito di svariati test si può notare come i dati dei bambini DSA siano in numero troppo inferiore rispetto ai dati non DSA.

sottometodo	1	1
learnig_rate	10^{-4}	10^{-4}
batch_size	8	32
epoches	300	300
pre-learned 1	70,84%	81,42%
pre-learned 2	86,19%	44,24%
pre-learned 3	71,03%	47,14%
LOSS.	ACC.	ACC.
TEST	TEST	TEST

Tabella 4.9: Accuratezza Validation rete CNN sui bambini, sottometodo 1
Modifiche al batch size per cercare delle configurazioni con accuratezza maggiore.

Infatti l'accuratezza resta bassa, i valori di recall e di precision ci fanno capire che ancora sono necessari ulteriori dati per poter parlare di una soluzione da prendere in considerazione come supporto alle decisioni di un esperto in materia di DSA. Dalla Tabella 4.9 si può notare che i valori di accuratezza e di loss non ci permettono di capire come vengono commessi tutti questi errori. Si ricorre per un analisi maggiore alla visualizzazione delle Figure 4.8 e 4.9.

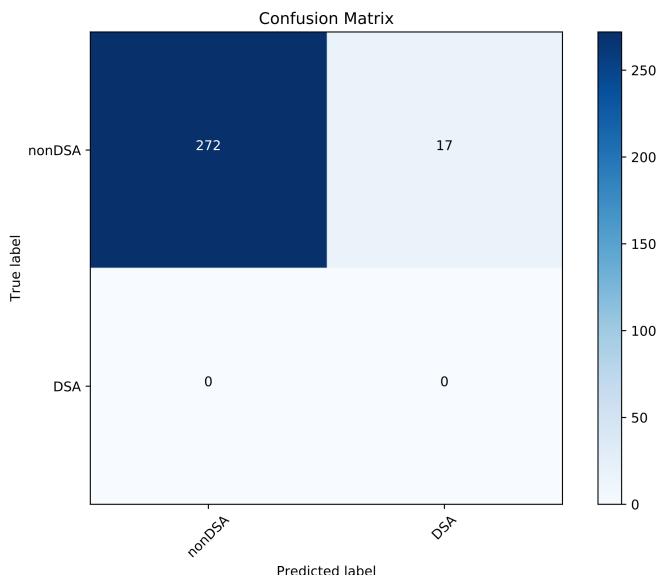


Figura 4.8: Matrice di confusione ottenuta con impostazioni prima colonna in Tabella 4.9

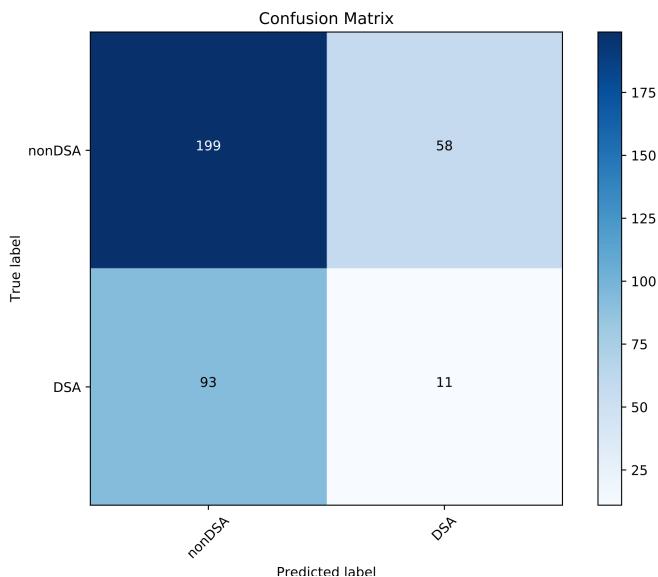


Figura 4.9: Matrice di confusione ottenuta con impostazioni seconda colonna in Tabella 4.9

Capitolo 5

Conclusioni e sviluppi futuri

Conclusioni

Fine delle fini, finalmente.

Il progetto ha avuto dei risultati, e questa è la parte importante. In generale non esistono risultati soddisfacenti o meno, ogni risultato porta una conoscenza che va sfruttata per studi futuri e possibili applicazioni pratiche. In questo progetto di tesi abbiamo analizzato la scrittura della mano principale e secondaria di studenti universitari come primo approccio al problema di riconoscere un bambino DSA a partire da un documento scritto. Si sono sviluppate tre tecniche con le quali abbiamo analizzato i testi prodotti dagli studenti universitari. Una di queste tecniche è stata implementata '*from scratch*' basandosi su di un test chiamato BHK largamente utilizzato in ambito di DSA. Le altre due tecniche utilizzano le Reti Neurali MLP e convoluzionali CNN per effettuare un'analisi alternativa al precedente test. Dai risultati ottenuti nel Capitolo 4 si nota come le prime due tecniche risultino meno accurate rispetto a quanto riesce a fare la Rete Neurale CNN, che con un'accuratezza massima dell' 87% riconosce la scrittura prodotta con la mano principale da quella con la mano secondaria. La seconda parte del

progetto ha visto coinvolta la scuola media del Comprensivo di Scarperia e San Piero che ci ha permesso di raccogliere documenti scritti da un centinaio di bambini. Su questi dati sono state applicate nuovamente le tecniche sopra discusse, ottenendo risultati meno positivi e meno accurati rispetto ai precedenti. Infatti questi tipi di reti (intendendo le Reti convoluzionali) lavorano bene laddove anche l'occhio umano riesce a lavorare bene, mentre i dati dei bambini sono molto variegati, poco uniformate le differenze fra le classi e non visibilmente distinguibili neanche da un umano.

Sviluppi futuri

Uno studio futuro potrebbe partire dal creare un dataset con le scritte dei bambini di 7 anni, infatti sono all'inizio dell'apprendimento della scrittura e le differenze nei modi di scrivere potrebbero risultare una componente importante per questo tipo di studi. Si potrebbe poi pensare a migliorare il primo metodo di estrazione delle feature, quello sviluppato nella prima parte del progetto, poichè tutt'ora non si tiene traccia della cronologicità dei tratti durante gli esperimenti, o di parametri come il *time-stamp* e la *pressure*.

Altre tecniche di analisi della scrittura che potrebbero essere interessanti sono tutte quelle che stanno sotto il nome di *Analisi Online*. Infatti vi sono LSTM, BLSTM, Hidden Markov Model, e molte altre tecniche che non si ha avuto modo di apporfiondire in questa tesi ma che potrebbero essere strade valide per sviluppi futuri.

Un ultimo argomento di cui sono venuto a conoscenza durante la conclusione del progetto riguarda un metodo diverso di estrazione dei dati dalla scrittura a mano che consiste nell'utilizzo di un anello indossabile composto da molteplici sensori che rileva i movimenti della mano e delle dita dello scrittore mentre produce del testo [35]. Questo strumento potrebbe essere

frutto anch'esso di futuri approfondimenti sia per studiare nuove tecnologie, sia per sfruttarle con l'obiettivo di distinguere i bambini DSA dai non DSA.

Ringrazio i lettori per l'attenzione, con la speranza di averli incuriositi sull'argomento e aver stimolato la fantasia su queste tematiche di beneficio della collettività.

Elenco delle figure

1.1	Elenco test DSA.[6]	4
1.2	Esempio di applicazione del test BHK.	8
1.3	Esempio di applicazione del test BVMGT [12].	10
1.4	Rappresentazione di un neurone biologico ed uno artificiale.[15]	12
1.5	Percettrone vs. Percettrone multistrato.	15
1.6	CNN per riconoscimento di caratteri numerici [17].	16
1.7	Schema di funzionamento schermi passivi.[5]	21
1.8	Schema di funzionamento tecnologie attivi.[23]	23
2.1	Esempio di scheletro e di contorno [24].	26
2.2	Albero composizione file WILL.	27
2.3	Modello concettuale dell'oggetto Path	28
2.4	Oggetto Path e algoritmo Catmul-Rom	28
2.5	Lista a lughezza variabile degli stroke codificati	30
2.6	Creazione delle parole unendo gli stroke	34
2.7	Creazione delle righe unendo le parole consecutive	34
2.8	Grandezza della scrittura	35
2.9	Margine sinistro non allineato	36
2.10	Andamento altalenante della linea di scrittura	37
2.11	Spazio insufficiente tra le parole e tra le lettere	38
2.12	Lettere ritoccate o ricalcate	38

2.13	Feature "grezze" prelevate fra le righe con BHK modificato . . .	43
2.14	Feature "grezze" prelevate fra le parole con BHK modificato . .	43
3.1	Esempio di documenti prodotti da studenti universitari.	49
3.2	Bambino di 11 anni mentre esegue il test sottoposto	50
3.3	Esempio di documenti prodotti da bambini di prima media. . .	50
4.1	Gaussiane dei valori bambini non DSA (arancione) e DSA (blu)	60
4.2	Matrice di confusione mediata su k , dataset universitari (100 epoch)	62
4.3	Matrice di confusione mediata su k , dataset universitari (300 epoch)	63
4.4	Matrice di confusione mediata su k , dataset medie (100 epoch)	64
4.5	Matrice di confusione mediata su k , dataset medie (300 epoch)	64
4.6	Rappresentazione AlexNet della CNN utilizzata prima della modifica (50 nodi di output invece di 2)[34]	66
4.7	Rappresentazione LetNet della CNN utilizzata prima della modifica (50 nodi di output invece di 2)[34]	67
4.8	Matrice di confusione ottenuta con impostazioni prima colon- na in Tabella 4.9	72
4.9	Matrice di confusione ottenuta con impostazioni seconda co- lonna in Tabella 4.9	72

Elenco delle tabelle

2.1	Caratteristiche della pagina	41
2.2	Caratteristiche della riga	41
4.1	Primo Dataset	57
4.3	Analisi dataset scuola media	58
4.2	Analisi dataset scuola media	59
4.4	Media delle accuratezze sul test (100 epoche) con deviazione standard, dataset universitari	62
4.5	Media delle accuratezze sul test (300 epoche) con deviazione standard, dataset universitari	63
4.6	Media delle accuratezze sul test (100 epoche) con deviazione standard, dataset medie	64
4.7	Media delle accuratezze sul test (300 epoche) con deviazione standard, dataset medie	64
4.8	Accuratezza Validation rete CNN sugli universitari, sottometodo 1 vs. 2	70
4.9	Accuratezza Validation rete CNN sui bambini, sottometodo 1	71

Bibliografia

- [1] Zappaterra Tamara and Trisciuzzi Leonardo. La dislessia. <https://www.ibs.it/dislessia-libro-tamara-zappaterra-leonardo-trisciuzzi/e/9788881073689>, 2014.
- [2] Flavia Crescenzi, Simonetta Rossi, Francesca Aurigemma, Carmelina Cammisa, Donatella Dormi, Luisa Lopez, and Roberta Penge. I dsa a scuola. una guida per gli insegnanti. <http://www.libriliberi.com/prodotto/i-dsa-a-scuola-una-guida-per-gli-insegnanti/>.
- [3] Giacomo Stella and Luca Grandi. Come leggere la dislessia e i dsa. <http://www.giunti.it/libri/educazione-e-scuola/come-leggere-la-dislessia-e-i-dsa2/>.
- [4] Claudio Vio, Patrizio E. Tressoldi, and Gianluca Lo Presti. Diagnosi dei disturbi specifici dell'apprendimento scolastico. <https://www.erickson.it/Libri/Pagine/Scheda-Libro.aspx?ItemId=37394>, 2012.
- [5] Wikipedia.

- [6] Training Cognitivo. La valutazione della dislessia, i test. [http://www.trainingcognitivo.it/la-valutazione-della-dislessia-i-test//](http://www.trainingcognitivo.it/la-valutazione-della-dislessia-i-test/).
- [7] Carlo Di Brina and Giovanna Rossini. Test bhk - scala sintetica per la valutazione della scrittura in età evolutiva. 2011.
- [8] Michela Borean, Giulia Paciulli, Laura Bravar, and Stefania Zoia. Test dgm-p - test per la valutazione delle difficoltà grafo-motorie e posturali della scrittura. 2012.
- [9] Patrizio E. Tressoldi, Cesare Cornoldi, and Anna Maria Re. Bvsco-2 - batteria per la valutazione della scrittura e della competenza ortografica. 2013.
- [10] Lauretta Bender. Bender visual motor gestalt test. 1979.
- [11] Martina Cinguetti. Bhk: strumento di valutazione della grafia. <http://logos-martina.blogspot.com/2013/04/bhk-strumento-di-valutazione-della.html>, 2013.
- [12] Dr. Harold Levinson. Esempio di applicazione del test bvmgt, sito dislesia online. 2018.
- [13] Robert Hecht-Nielsen. Pioneer in artificial-intelligence software devises new theory of cognition. 1947.
- [14] Fumo David. A gentle introduction to neural networks series. <https://towardsdatascience.com/a-gentle-introduction-to-neural-networks-series-part-1-2b90b87795bc>, 2017.

- [15] Sconosciuto. Rappresentazione di un neurone biologico ed uno artificiale. (da mettere il link). <https://??>, 2018.
- [16] Sefik Ilkin Serengil. Percettrone vs. percettrone multistrato. (da mettere il link). 2018.
- [17] Sefik Ilkin Serengil. Riconoscimento di caratteri numerici usando cnn con keras. <https://sefiks.com/2017/11/05/handwritten-digit-recognition-using-cnn-with-keras/>, 2018.
- [18] R. Plamondon and S.N. Srihari. Online and off-line handwriting recognition: a comprehensive survey. <https://ieeexplore.ieee.org/document/824821>, 2000.
- [19] V. Bouletreau, N. Vincent, R. Sabourin, and H. Emptoz. Handwriting and signature: One or two personality identifiers? *Proc. 14th Int'l Conf. Pattern Recognition*, (1,758):2, 1998-Aug.
- [20] Google Inc. Google books library project. <https://books.google.com/googlebooks/library>, 2004.
- [21] Muhammad'Arif Mohamad, Dewi Nasien, Haswadi Hassan, and Habibollah Haron. A review on feature extraction and feature selection for handwritten. (*IJACSA*) *International Journal of Advanced Computer Science and Applications*, 6(2):9, 2015.
- [22] Giacomo D. Fumagalli. I digitalizzatori. <http://www.tabletpc.it/it/cosa-sono-i-digitalizzatori/>.
- [23] ElectronicDesign. The year of the digital pen. <https://www.electronicdesign.com/blog/year-digital-pen>, 2013.

- [24] Stefan Pletschacher, Marcel Eckert, and Arved C. Huebler. Vectorization of glyphs and their representation in svg for xml based processing. pages 299–308, 01 2006.
- [25] Developer documentation Wacom. Descrizione schema classe path, sezione path data. <https://developer-docs.wacom.com/display/DevDocs/WILL+File+Format/>, 2018.
- [26] Inkspace developer Wacom. Pagina ufficiale wacom inkspace. <https://www.wacom.com/it/getting-started/inkspace>, 2018.
- [27] Inkspace developer Wacom. Piattaforma inkspace per la visione dei documenti digitali prodotti con tecnologia wacom. <https://inkspace.wacom.com/?stream=inkspace>, 2018.
- [28] Emanuele Vivoli. Repository del progetto sulla piattaforma github. https://github.com/emanuelevivoli/Thesis_BHK, 2018.
- [29] François Chollet et al. Keras. <https://keras.io>, 2015.
- [30] Martín Abadi et. al. TensorFlow: Large-scale machine learning on heterogeneous systems. <https://www.tensorflow.org/>, 2015. Software available from tensorflow.org.
- [31] UV. Marti and H. IJDAR Bunke. The IAM-database: an english sentence database for offline handwriting recognition. <https://doi.org/10.1007/s100320200071>, 2002.
- [32] Dwivedi Priya. English writer identification. https://github.com/priya-dwivedi/Deep-Learning/blob/master/handwriting_recognition/English_Writer_Identification.ipynb, 2018.

- [33] Linjie Xing and Yu Qiao. Deepwriter: A multi-stream deep CNN for text-independent writer identification. *CoRR*, abs/1606.06472, 2016.
- [34] Lenail Alex. Publication-ready nn-architecture schematics. <http://alexlenail.me/NN-SVG/>, 2018.
- [35] Jing Lei, Dai Zeyang, and Zhou Yiming. Wearable handwriting recognition with an inertial sensor on a finger nail. <https://ieeexplore.ieee.org/document/8270150/>, 2017.