STATICO VS. DINAMICO.
UN POSSIBILE RUOLO DELLA SILLABA
NEL RICONOSCIMENTO AUTOMATICO DEL PARLATO

Serena Soldo, Bogdan Ludusan
Università degli studi di Napoli "Federico II"
soldo@na.infn.it, ludusan@na.infn.it

1. SOMMARIO

Il presente lavoro si pone come obiettivo quello di esplorare possibili tecniche di rappresentazione delle sillabe. La proposta è quella di utilizzare le caratteristiche statiche del segnale che rappresenta una sillaba. I parametri per questo tipo di rappresentazione sono stati estratti secondo due tecniche diverse: usando un numero variabile di parametri oppure un numero fisso. Per scegliere il tipo di rappresentazione migliore è stato addestrato un classificatore SVM. Le prestazioni migliori sono state ottenute utilizzando 15 frames per sillaba, ciascuno rappresentato con 13 parametri MFCC, raggiungendo un'*accuracy* pari a 88,25%.

2. INTRODUZIONE

Il continuum fonico su cui un sistema automatico di riconoscimento deve lavorare viene normalmente segmentato in piccole porzioni sulle quali algoritmi basati su tecniche statistiche operano sia per l'identificazione dell'informazione linguistica in essi contenuta, sia per ricostruire a posteriori il contenuto complessivo dell'enunciato contenuto nel segnale acustico. Mentre tradizionalmente fino a pochi anni fa le dimensioni della porzione minima di analisi si aggiravano intorno a dimensioni che linguisticamente potremmo definire subfoniche, sempre più spesso, ormai, i sistemi di riconoscimento del parlato fanno uso di analisi di segmenti di parlato superiori ai 150-200 ms. Questa tendenza indica l'uso di parametri soprasegmentali oltre che segmentali. Fra i parametri per la descrizione di segmenti lunghi che è possibile usare si incontrano quelli legati a proprietà ritmiche del parlato. Recentemente sono stati portati avanti lavori per dimostrare che tali parametri possono essere estratti automaticamente con algoritmi indipendenti dalla lingua (Tamburini & Caini, 2005; Petrillo, 2000; Ludusan & Soldo, 2009).

Sebbene la definizione 'classica' di sillaba (ma i linguisti sanno bene quanto trovare una definizione condivisa da tutti sia difficile) solitamente utilizzata in letteratura tende a mettere in evidenza le caratteristiche dinamiche del segnale vocale come ad esempio la coarticolazione, in questo lavoro si è cercato di proporre una ipotesi alternativa. L'idea è quella di vedere la sillaba come una rappresentazione statica di un pezzo di parlato, una sorta di istantanea che contenga in sé unitariamente informazione che solitamente si ritiene di tipo tempo-variabile, che si estende su un determinato intervallo di tempo. Alla luce di questo tipo di rappresentazione, la variabile indipendente rispetto alla quale i fenomeni che osserviamo evolvono e sulla quale possiamo basare un sistema di riconoscimento del parlato non risulterà più essere il tempo, ma la sequenza di unità sillabiche. Supponendo di essere in grado di individuare con precisione gli estremi dell'intervallo su cui si estende ciascuna sillaba, si può dunque pensare di 'fotografarla' estraendone le caratteristiche nei punti salienti.

È da osservare che questo genere di rappresentazione della sillaba è completamente originale e mai proposto in letteratura. Lo scopo di questo lavoro è proprio quello di capire se si tratta di una tecnica in grado di fornire buoni risultati e, eventualmente, di evidenziarne i punti deboli.

3. STRUMENTI

3.1 Support Vector Machine e LIBSVM

Tra i classificatori supervisionati più noti in letteratura troviamo le cosiddette *Support Vector Machines* (SVM) (Boser *et al.*, 1992; Cortes & Vapnik, 1995; Vapnik, 1995). In genere l'uso più comune, e per il quale le SVM risultano particolarmente adatte, è la classificazione di oggetti appartenenti a due sole classi ma esse possono essere facilmente estese anche al caso di più classi. Inoltre, negli ultimi anni sono stati tentati approcci nell'uso delle SVM proprio per la classificazione del parlato (in particolare Ganapathiraju, 2002). Le SVM fanno parte della famiglia di classificatori a 'massimo margine', infatti hanno come scopo quello di individuare una superficie di decisione che sia il più lontana possibile da ciascun punto dell'insieme dei dati. Tale distanza rappresenta il 'margine' del classificatore. Il nome *Support Vector Machines* deriva dal ruolo fondamentale di un particolare insieme di dati chiamati 'vettori di supporto'. Questi vettori sono, in realtà, gli unici ad avere peso nella scelta della superficie di decisione. In figura 1 è mostrato un esempio di superficie di decisione (a sinistra) e la superficie di decisione ottima (a destra) con i corrispondenti vettori di supporto. Osserviamo che le SVM sono l'unica famiglia di classificatori che può garantire determinate prestazioni di generalizzazione. Infatti la teoria di Vapnik (Vapnik, 1995) dimostra che la soluzione a massimo margine è anche la soluzione a massima generalizzazione.

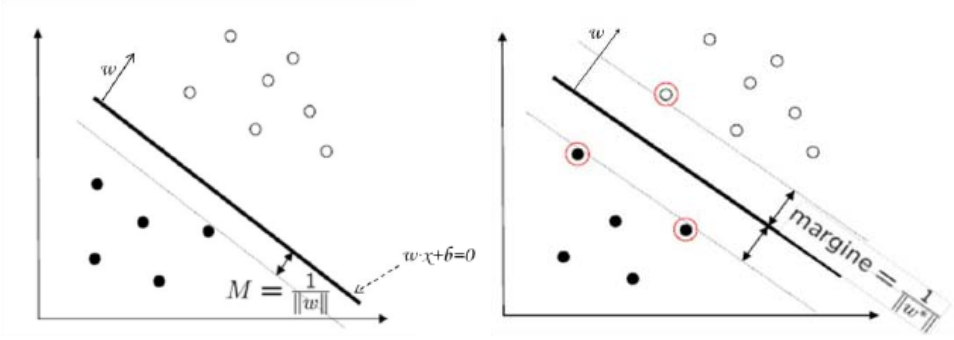

Figura 1: Esempio di un generico iperpiano di decisione (a sinistra) e l'iperpiano di decisione ottimo (a destra) per un problema di classificazione basato su due classi. Gli elementi cerchiati rappresentano i vettori di supporto

Formalmente possiamo definire il problema di classificare tramite SVM nel seguente modo: sia S = (x_1; y_1), (x_2; y_2), …(x_n; y_n) un training-set di punti tale che $x_i \in \Re^m$ e

$y_i \in \{-1; +1\}$ e sia D(x) := ω·x+b = 0 l'equazione per un generico iperpiano, si vuole risolvere il seguente problema di ottimizzazione:

$$\begin{aligned} &\min_{\omega,b} && \tfrac{1}{2}|\omega|^2 \\ &\text{s. a} && \\ &&& y_i(\omega \cdot x_i + b) \geq 1, \quad \forall i = 1...n \end{aligned} \tag{1}$$

La soluzione, ovvero il margine massimo, sarà data da $M^* = 1/|\omega^*|$. Tale soluzione può essere individuata con uno qualsiasi degli algoritmi ideati per i problemi di ottimizzazione con funzione obiettivo quadratica e vincoli lineari.

Nel caso in cui il training-set non risulti linearmente separabile, è possibile proiettarlo in un spazio di dimensione maggiore in cui aumenta la possibilità che i punti siano linearmente separabili (Cover, 1965). Le funzioni di Kernel hanno lo scopo simulare la proiezione dei punti in un nuovo spazio nel caso essi non siano linearmente separabili. Le funzioni di Kernel devono essere funzioni continue, simmetriche e definite positive. Tra le più usate funzioni di Kernel ci sono le funzioni a base radiale e le funzioni polinomiali.

Per la costruzione di un classificatore sillabico tramite SVM è stata usata una libreria specifica, LIBSVM (Chang & Lin, 2001), che mette a disposizione funzioni per il *training* del classificatore, lo *scaling* dei parametri, e la classificazione di nuovi oggetti.

La libreria è molto flessibile e permette l'uso di diversi kernel e la personalizzazione dei relativi parametri. Per quanto riguarda la classificazione 'multi-class' (ovvero con più di due classi), la libreria implementa la tecnica 'uno-contro-uno' poiché gli esperimenti degli autori (Hsu *et al.*, 2003) hanno mostrato che nonostante il numero di classificatori binari utilizzato sia maggiore, le prestazioni in termini di tempo e di risultati ottenuti sono decisamente migliori. Come vedremo nel paragrafo 4, dai test effettuati è emerso che, tra i kernel messi a disposizione dalla libreria, il kernel *Radial Basis Function (RBF)* è il più efficace per la classificazione del nostro spazio di dati. La caratteristica di questo kernel è quella di saper modellare aree di decisione chiuse. Questa proprietà è utile quando l'insieme dei dati si distribuisce nello spazio in modo che una classe sia completamente incapsulata in un'altra. Evidentemente la grossa somiglianza tra alcune coppie di sillabe (ad esempio 'di' e 'dje' oppure 'tre' e 'tren') rende fondamentale per la classificazione l'uso di un kernel come quello RBF.

3.2 Il corpus

Il corpus adottato in questo lavoro è una parte del corpus SPEECON (Siemund *et al.*, 2000); in questo corpus sono presenti 18 differenti lingue. La parte del corpus utilizzata riguarda i numeri in lingua italiana tra 0 e 999,999 pronunciati da soli speaker maschi; ogni registrazione audio ha una frequenza di 16 KHz e contiene l'enunciazione di un qualsiasi numero tra 0 e un milione (escluso). Le registrazioni audio sono 1906, pronunciate da circa 400 speaker differenti, i quali hanno registrato in media circa 5 file a testa. In base alla divisione sillabica fonologica (Cutugno *et al.*, 2001), gli enunciati contenuti nei file del corpus sono stati suddivisi in 8631 sillabe distinte che vanno a coprire l'intero insieme di 42 classi di sillabe presenti nel corpus. La tabella 1 mostra l'elenco delle sillabe e le relative occorrenze all'interno del corpus. Osserviamo che la sillaba 'due' costituisce un caso

particolare. Secondo Canepari "la distinzione tradizionale fra dittongo e iato è puramente teorica" (Canepari, 1999: 143). Se è vero che per [due] vs. [dwe] in forma isolata possono sussistere dei dubbi, è indubbio lo spostamento di accento in tutti i casi in cui il numero inizia con 'due' e segue: in questi casi 'due' cliticizza ed è sempre una sillaba.

Sillaba	**# occ.**	**Sillaba**	**# occ.**	**Sillaba**	**# occ.**
di	361	no	462	to	614
dje	67	o	177	tre	259
do	57	ran	62	tren	64
due	233	ro	119	ttan	111
dze	121	se	391	tte	283
sei	220	ssan	87	tto	293
kwa	360	sse	62	ttor	52
kwan	86	tʃa	114	ttro	246
kwe	215	tʃen	581	tu	53
kwin	50	tʃi	314	u	123
la	300	tʃin	300	un	57
lle	50	tʃo	55	van	57
mi	356	ta	367	ve	267
nno	52	ti	54	ven	61

Tabella 1: Elenco delle sillabe e relativo numero di occorrenze all'interno del *corpus*

4. METODO

4.1 Primo approccio: la sillaba come un volto

Il primo passo del nostro lavoro è consistito nella trasformazione di ogni porzione di segnale corrispondente ad una sillaba in un set di parametri (d'ora in poi *features*) da fornire in ingresso ad un sistema di riconoscimento.

Come è noto, una sillaba è costituita da almeno una vocale (che ne costituisce il nucleo) e può al massimo essere formata da tre parti, il nucleo vocalico testé definito, la testa e la coda. Per rappresentare ciascuna sillaba abbiamo quindi scelto di concentrare l'estrazione delle *features* solo sul centro di ciascuna delle tre parti. In particolare sono stati estratti tre vettori di parametri per la testa, tre per il nucleo e tre per la coda. Si è scelto di utilizzare per la rappresentazione i 13 coefficienti MFCC (*Mel Frequency Cepstral Coefficients*). Ciascuna sillaba, alla luce di queste scelte, risulta essere rappresentata da una matrice di dimensioni 9 x 13.

L'idea di rappresentare la sillaba in questo modo è nata dall'incontro con tecniche simili utilizzate nell'ambito del riconoscimento dei volti (ad esempio Samaria & Fallside, 1993). I tratti somatici si presentano sempre nello stesso ordine, indipendentemente dall'angolazione in cui si presenta il volto. Più precisamente, un viso può sempre essere diviso in 5

fasce orizzontali che individuano: Fronte, Occhi, Naso, Bocca, Mento. La nostra rappresentazione segue questo stesso principio spezzando ciascuna sillaba nei suoi 'tratti somatici' (testa, nucleo e coda) ed estraendo informazioni per ciascun segmento. La figura 2 mostra un esempio di tale suddivisione per un volto e per una sillaba.

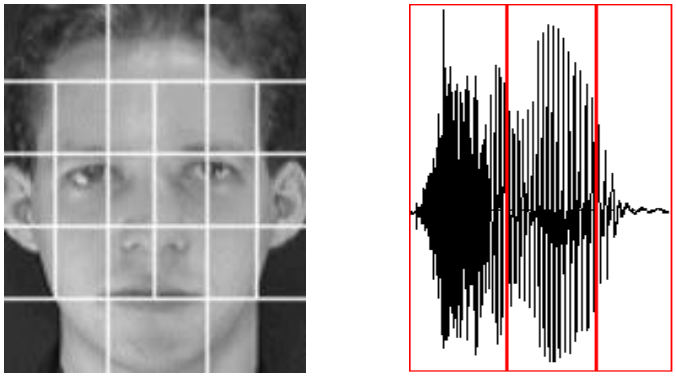

Figura 2: Esempio di segmentazione di un volto (a sinistra)
e di una sillaba (a destra) nei rispettivi 'tratti somatici'

Scelto questo tipo di rappresentazione, abbiamo addestrato una SVM multiclasse sulla base di un *training set* estratto dal corpus. In particolare l'SVM è stato addestrato sulle 42 classi di sillabe fonologiche presenti nel corpus e su una ulteriore classe che comprende il silenzio e le eventuali zone rumorose. L'efficacia della rappresentazione è stata poi valutata sul test set. Le prestazioni ottenute dalla classificazione delle sillabe rappresentate con questo primo approccio è pari all'85% circa (i risultati in dettaglio sono riportati nel § 4).

4.2 Evoluzioni

Le prestazioni ottenute dal classificatore utilizzando il metodo descritto al paragrafo precedente sono risultate incoraggianti anche se non particolarmente elevate a causa del fatto che la scelta di soli nove *frames* per la rappresentazione di qualsiasi sillaba non è forse la più adatta. Quindi una delle ipotesi prese in considerazione è stata quella di cambiare il numero di *frames* utilizzati. Le possibili tecniche da utilizzare a questo scopo sono due: il numero di *frames* considerati per ciascun segmento può variare al variare della dimensione del segmento stesso; oppure il numero di *frames* può essere fissato a priori ma con un valore più alto, scegliendo una maggiore o minore sovrapposizione delle finestre di avanzamento durante l'estrazione delle *features* in modo da adattarsi a ciascun segmento. Nella realizzazione della prima tecnica la sovrapposizione tra le finestre durante l'estrazione delle *features* è stata mantenuta fissa e quindi per ogni segmento sono stati estratti un numero di vettori variabile in base alla lunghezza del segmento stesso. Per questa tecnica sono state fatte delle prove considerando finestre di 128, 256 e 512 campioni (corrispondenti a circa 8, 16 e 32 msec), ma entrambe hanno prodotto risultati molto scarsi; per tale motivo questo sistema è stato accantonato in favore della seconda tecnica, molto più promettente. La seconda tecnica mira ad ottenere un numero di *frames* fisso da ciascun segmento; questo è stato ottenuto fissando l'ampiezza della finestra (nel nostro caso 256 campioni, corrispondenti a 16 msec) e variando l'ampiezza della sovrapposizione opportunamente.

5. RISULTATI

Le due tecniche di rappresentazione delle sillabe sono state testate tramite un classificatore SVM opportunamente addestrato. Le tabelle 2 e 3 riportano nel dettaglio i valori di *accuracy* ottenuti per ciascuna variante.

Shift tra i *frames* (ms)	Kernel Lineare [%]	Kernel Polinomiale [%]	Kernel RBF [%]
8	78,11	74,06	65,50
16	77,61	73,70	65,46
32	75,13	71,09	63,92

Tabella 2: Prestazione della classificazione per i diversi tipi di kernel utilizzando un numero variabile di frames per ogni sillaba

Numero di *frames* per sillaba	Kernel Lineare [%]	Kernel Polinomiale [%]	Kernel RBF [%]
15	86,53	85,88	88,25
17	85,85	85,63	88,10
19	85,95	85,60	88,21
21	85,99	85,81	88,75
23	85,81	85,56	88,18
25	86,10	85,99	88,32

Tabella 3: Prestazione della classificazione per i diversi tipi di kernel utilizzando un numero fisso di frames per ogni sillaba

Come si può notare, l'uso di un numero di *frames* variabile si è dimostrata una tecnica completamente inefficace mentre l'uso di un numero fisso di *frames* per sillaba è risultato decisamente migliore. In particolare, abbiamo fatto variare il numero di *frames* per ogni sillaba tra 15 e 25; le prestazioni ottenute all'aumentare del numero di *frames* non sono risultate significativamente migliori. Per tale motivo abbiamo fissato a 15 il numero di *frames* ideale per questo genere di rappresentazione.

Un'ulteriore analisi che si può fare sui risultati della classificazione è la valutazione degli N-best. Un classificatore SVM, nel tentativo di predire la classe da attribuire ad un elemento, individua la probabilità di appartenenza dell'elemento stesso ad ogni possibile classe: la classe con probabilità più alta è quella restituita in output. Facendo in modo che la SVM restituisca in output anche tutte le coppie <classe, probabilità> per ogni elemento da classificare e per ogni classe, è possibile valutare in che posizione di questa 'classifica' si trova la classe corretta di appartenenza di ciascun elemento. La tabella 4 riassume i risultati emersi da questa analisi. In particolare si osserva che nel 96% dei casi la classe giusta rientra tra le prime 3 più probabili e nel 99% dei casi essa è nelle prime 10.

N-best	*Accuracy* [%]
1	88.25
3	96.67
5	98.14
10	99.18
30	99.96

Tabella 4: Percentuali di *accuracy* nella valutazione degli N-best

6. DISCUSSIONE

I risultati ottenuti incoraggiano la costruzione di un algoritmo di decodifica che combini le informazioni fornite dal classificatore con quelle 'top down' provenienti dal dizionario usato per il riconoscimento fornendo in output la sequenza di sillabe che ha più probabilità di essere contenuta nel segnale da riconoscere.

In conclusione questo lavoro ha indagato nuove tecniche di estrazione delle features per la rappresentazione delle sillabe. Abbiamo mostrato come un approccio teso a estrarre informazioni sulle caratteristiche statiche del segnale, piuttosto che quelle dinamiche, può fornire buoni risultati.

7. BIBLIOGRAFIA

Boser, B.E., Guyon, I.M., & Vapnik, V.N. (1992), A training algorithm for optimal margin classifiers, in *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, Pittsburgh, Pennsylvania, July 27-29, 1992, 144-152.

Canepari, L. (1999), *Manuale di Pronuncia Italiana*, Bologna: Zanichelli.

Chang, C.C. & Lin, C.J. (2001), LIBSVM: a library for support vector machines, Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Cortes, C., & Vapnik, V. (1995), Support vector networks, *Machine Learning*, 273-297.

Cover, T.M. (1965), Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition, *IEEE Transactions on Electronic Computers*, 14(3), 326-334.

Cutugno, F., Passaro, G. & Petrillo, M. (2001), Sillabificazione fonologica e sillabificazione fonetica, in *Dati empirici e teorie linguistiche*, Atti del XXXIII Congresso della Società di Linguistica Italiana, Napoli, 28-30 ottobre 1999 (F. Albano Leoni, R. Sornicola, E. Stenta Krosbakken, C. Stromboli, editors), Roma: Bulzoni, 205-232.

Ganapathiraju, A. (2002), *Support Vector Machines for Speech Recognition*, PhD thesis, Faculty of Mississippi State University.

Hsu, C.-W., Chang, C.-C., & Lin, C.-J. (2003), *A practical guide to support vector classification*, Technical report, Taipei, Taiwan: National Taiwan University.

Ludusan, B. & Soldo, S. (2009), Sonority based syllable segmentation, in *La dimensione temporale del parlato* (S. Schmid, M. Schwarzenbach & D. Studer, editors), Atti del 5° Convegno Nazionale dell' Associazione Italiana di Scienze della Voce, Zurigo, Svizzera, 4-6 febbraio 2009, 699-706 (in questo volume).

Petrillo, M. (2000), *Algoritmi per la divisione del segnale verbale in unità sillabiche*, Tesi di Laurea presso l'Università degli Studi Di Napoli "Federico II".

Samaria, F. & Fallside, F. (1993), Face Identification and Feature Extraction Using Hidden Markov Models, in *Image Processing: Theory and Applications*, 1, (G. Vernazza, editor), Amsterdam: Elsevier, 295-298.

Siemund, R., Höge, H., Kunzmann, S., & Marasek, K. (2000), Speecon - Speech data for consumer devices, in *Proceedings of International Conference on Language Resources and Evaluation (LREC)*, Athens, Greece, May 31-June 2, 2000, 883-886.

Tamburini, F. & Caini, C. (2005), An Automatic System for Detecting Prosodic Prominence in American English Continuous Speech, *International Journal of Speech Technology 2005*, 8, 33 – 44.

Vapnik, V.N. (1995), *The Nature of Statistical Learning Theory*, New York: Springer.