Traitement Automatique de la parole

La correction est en italique.

Reconnaissance des chiffres du français.

La tâche de reconnaissance envisagée est la reconnaissance de chiffres. Le vocabulaire de l'application se compose de 10 mots.

De manière très traditionnelle, le signal acoustique est analysé par fenêtre de 20 ms, toutes les 10 ms. Ce pré traitement consiste à attribuer à chaque trame de signal un code à partir du dictionnaire suivant

D = { X, C, U, V}

avec les interprétations suivantes :

V signifie que la trame est de type voyelle

X signifie que la trame est non voisée

C représente un silence

U représente une trame consonantique voisée.

En d'autres termes, après prononciation d'un chiffre, il est obtenu une suite d'observations $(O_1, O_2, ..., O_t, ..., O_T)$. A titre d'exemple, la prononciation du chiffre « 4 » peut produire la suite d'observations XVVCX.

Etant données les contraintes de temps de cet examen et les moyens de calcul limités, la tâche de reconnaissance pour la suite des questions, est restreinte au vocabulaire { 6, 7, 8}.

1 Reconnaissance de mots isolés par programmation dynamique.

On suppose dans cette question que la tâche est la reconnaissance de chiffres en mode isolé.

Rappelez le principe de la reconnaissance par programmation dynamique.

Proposez pour chacun des trois chiffres { 6, 7, 8} une suite d'observations (limitez le nombre d'observations à 5 pour chaque suite, n'insérez pas de silence au début et à la fin !). Ces trois suites sont considérées comme les prononciations de référence pour chacun des trois chiffres.

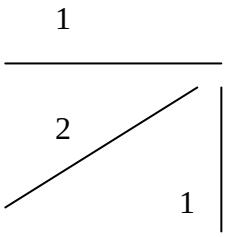
Une prononciation du mot « 6 » peut être / XVX/, elle sera noté R_6.
Une prononciation du mot « 7 » peut être / XVCX/, elle sera noté R_7.
Une prononciation du mot « 8 » peut être / UVCX/, elle sera noté R_8.

En phase de test, la prononciation d'un chiffre donne la suite d'observations XXVVUX. Après avoir défini des contraintes locales aussi simples que possible et compatibles avec l'algorithme de programmation dynamique (expliquez), appliquez l'algorithme de programmation dynamique sur vos références pour trouver quel mot est prononcé, en utilisant le tableau 1 des distances entre observations.

D(i,j)	V	X	C	U
V	0	2	2	1
X	2	0	1	1
C	2	1	0	1
U	1	1	1	0

tableau 1 : Distance entre deux observations.

Voir Cours : les contraintes locales sont choisies les plus simples, à savoir tout allongement ou réduction d'un son est possible, ce de manière équivalente sans pénalisation :



Seul le tableau permettant l'alignement entre R_6 et le mot inconnu est reporté. Les deux autres calculs sont identiques.

	X	X	V	V	U	X
X	2	2	2	2	2	1
V	2	2	0	0	1	3
X	0	0	2	4	5	5

On en déduit que (voir cours) $D(M, R_6) = \frac{g(I,J)}{I+J} = \frac{1}{3+6} = \frac{1}{9}$

Pour les autres, on trouve que

$$D(M, R_7) = \frac{2}{10}$$

$$D(M, R_8) = \frac{4}{10}$$

Le mot reconnu est donc « 6 », il correspond à la valeur de distance la plus faible.

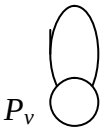
2 Reconnaissance de mots connectés par modèles de Markov cachés.

La tâche de reconnaissance consiste à reconnaître une suite de chiffres et tout chiffre peut suivre de manière équiprobable tout autre chiffre, sans pause entre eux.

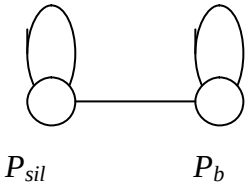
- Donnez l'ensemble des phonèmes qui interviennent dans la modélisation de 6, 7 et 8. Donnez le MMC (topologie, probabilités de transition, lois d'émission) pour chaque phonème intervenant dans la prononciation de 6, 7, 8. Compte tenu de l'ensemble d'observations, indiquez combien de MMC élémentaires sont différents et précisez combien de lois d'émissions différentes sont nécessaires pour les définir. Donnez les lois explicitement en remplissant le tableau 2 ; inspirez vous du tableau 1 et justifier votre démarche.

Les phonèmes sont / s i e t y / (y est la semi voyelle proche du son 'u')
Dans la mesure où les observations ne permettent pas de distinguer les voyelles entre elles, il n'y aura que 4 MMC élémentaires qui seront les suivants :

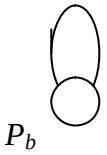
Modèle de la voyelle :



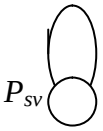
Modèle de la plosive sourde /t/ :



Modèle de la fricative /s/ :



Modèle de la semi voyelle :



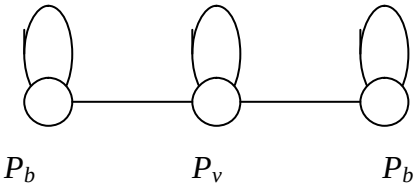
Pour écrire ces modèles, quatre lois d'émissions sont nécessaires, P_v caractérise une loi d'émission privilégiant l'étiquette V, P_{sil} l'étiquette C, P_b l'étiquette X et P_{sv} l'étiquette U. Des formes approchées de ces 4 lois sont données dans le tableau suivant(sous la forme –Log). La vraisemblance d'une observation est directement liée à la distance entre l'observation et l'étiquette attendue, d'où la similitude entre les deux tableaux (1 et 2).

	V	X	C	U
$-\log P_v$	0	2	2	1
$-\log P_{sil}$	2	1	0	1
$-\log P_b$	2	0	1	1
$-\log P_{sv}$	1	1	1	0

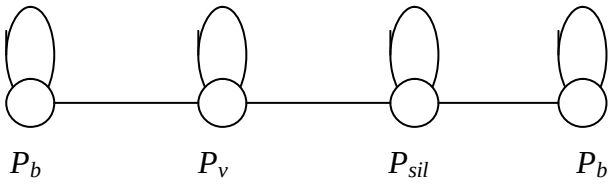
tableau 2 : Lois d'émission possibles pour les observations discrètes (V,U,X,C).

Construisez le modèle de chaque mot du vocabulaire (6,7,8), puis le modèle global (état, transition, lois) qui correspond à cette tâche en utilisant les MMC élémentaires précédemment définis. Factorisez ce modèle au maximum.

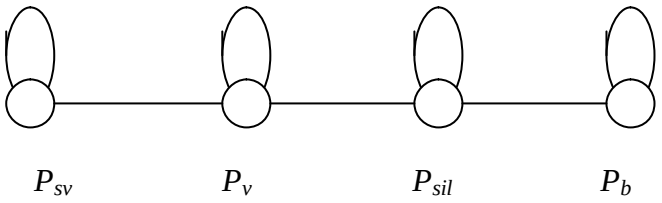
Mot « 6 » :



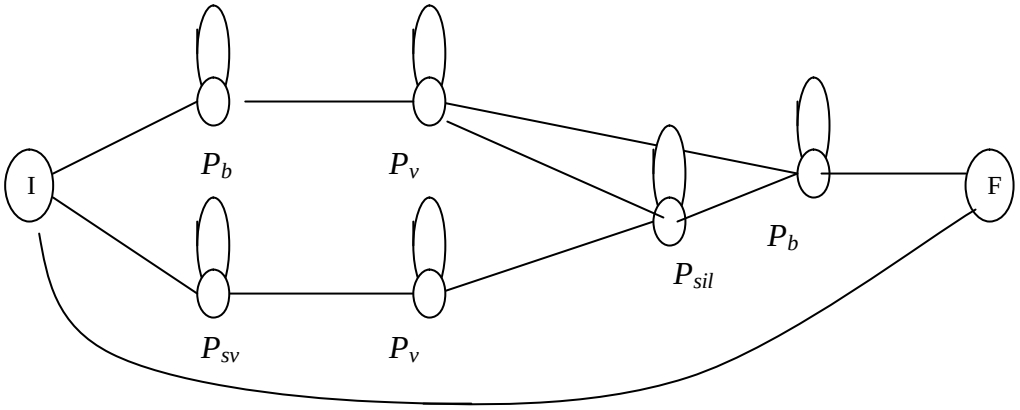
Mot « 7 »



Mot « 8 »



Modèle global factorisé :



Les deux états I et F sont introduits pour faciliter la construction mais ne portent pas de lois d'émissions.

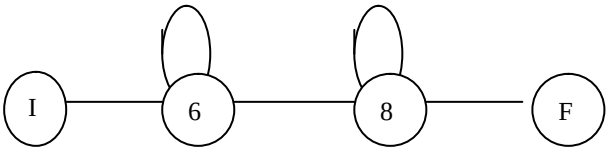
3 Reconnaissance de parole continue.

On se limite à la reconnaissance de suites de 6 et 8.

L'unité intermédiaire est désormais le mot. Supposez que les transitions internes à chaque mot sont telles que $-\log(a_{ij})=0$, et les transitions entre mots sont telles que $-\log(A_{uu'})=1$.

Pour chacun des deux mots, on reprend les modèles ci-dessus auxquels on ajoute un état de début et de fin formels, sans lois d'émissions. On numérote les états de la gauche vers la droite :
Le modèle du « 6 » est donc formé des états I_6 , 1 , 2 , 3 , F_6.
Le modèle du « 8 » est donc formé des états I_8 , 1 , 2 , 3 , 4 , F_8.
Les transitions internes sont telles que $-log(a_{ij})=0$.

Le modèle de langage est le suivant :



Les transitions de ce modèle sont telles que $-\log(A_{uu'})=1$.

Rappelez l'algorithme de Viterbi à deux niveaux et traduisez le sous forme –log (forme qui sera ensuite utilisée). En prenant les modèles de mots que vous avez défini en 2.2 pour 6 et 8, complétez chacun, si nécessaire, par un état initial et un état final formel (sans émission) ; posez le tableau qui vous permettra d'obtenir le meilleur chemin pouvant émettre la suite

d'observations XXVXUVCX en utilisant l'algorithme de Viterbi à deux niveaux (prenez une feuille entière !). Remplissez les quatre premières colonnes (la première correspond à l'initialisation du tableau sans émission d'observations).

	T=0	X	X	V	X	U	V	C	X
I_6	1	+∞	+∞	5+1=6	2	3	5	5	3
1	+∞	1	1	1+2=3	3+0=3	2+1=3	3+2=5	5+1=6	5+0=5
2	+∞	+∞	1+2=3	1+0=1	1+2=3	3+1=4	3+0=3	3+2=5	5+2=7
3	+∞	+∞	+∞	3+2=5	1+0=1	1+1=2	2+2=4	3+1=4	4+0=4
F_6	+∞	+∞	+∞	5	1	2	4	4	4
I_8	1	+∞	+∞	5+1=6	2	3	5	5	3
1	+∞	2	3	3+1=4	4+1=5	2+0=2	2+1=3	3+1=4	4+1=5
2	+∞	+∞	2+2=4	3+0=3	3+2=5	5+1=6	2+0=2	2+2=4	4+2=6
3	+∞	+∞	+∞	4+2=6	3+1=4	4+1=5	5+2=7	2+0=2	2+1=3
4	+∞	+∞	+∞	+∞	6+0=6	4+1=5	5+2=7	7+1=8	2+0=2
F_8	+∞	+∞	+∞	+∞	6	5	7	8	2
F	+∞	+∞	+∞	5+1=6	2	3	5	5	3

Le tableau est entièrement rempli, alors qu'il n'était demandé que les 5 premières colonnes (4 observations), uniquement pour que vous compreniez le fonctionnement global. Je n'ai pas indique toutes les flèches. Normalement il fallait toutes les stocker !! Le retour arrière indique que le meilleur chemin est (en le replaçant dans le bon ordre) I_6 1 1 2 3 F_6 I_8 1 2 3 4 F_8 F ce qui donne la suite de mots « 6 8 ».

A partir de la cinquième colonne, une heuristique d'élagage est appliquée « tout chemin partiel de probabilité dont la valeur –log est supérieure ou égale à un seuil, n'est pas prolongé ».

En prenant pour seuil 4, complétez le tableau. Donnez la suite de mots reconnue. Commentez le résultat.

	T=0	X	X	V	X	U	V	C	X
I_6	1	+∞	+∞	5+1=6	2	3	+∞	+∞	3
1	+∞	1	1	1+2=3	3+0=3	2+1=3	+∞	+∞	+∞
2	+∞	+∞	1+2=3	1+0=1	1+2=3	+∞	3+0=3	+∞	+∞
3	+∞	+∞	+∞	3+2=5	1+0=1	1+1=2	+∞	+∞	+∞
F_6	+∞	+∞	+∞	5	1	2	+∞	+∞	+∞
I_8	1	+∞	+∞	5+1=6	2	3	+∞	+∞	3
1	+∞	2	3	3+1=4	4+1=5	2+0=2	2+1=3	+∞	+∞
2	+∞	+∞	2+2=4	3+0=3	3+2=5	+∞	2+0=2	+∞	+∞
3	+∞	+∞	+∞	4+2=6	3+1=4	+∞	+∞	2+0=2	2+1=3
4	+∞	+∞	+∞	+∞	6+0=6	+∞	+∞	+∞	2+0=2
F_8	+∞	+∞	+∞	+∞	6	+∞	+∞	+∞	2
F	+∞	+∞	+∞	5+1=6	2	3	+∞	+∞	3

Dans ce cas cela n'a rien changé au résultat, cette technique permet d'accéder au résultat plus rapidement.

Ne serait il pas préférable de donner un seuil adaptatif ? Si oui, comment le choisiriez vous ?

Il est évident que ce seuil doit être adaptatif, car les scores augmentent, ne serait ce qu'à chaque passage par le modèle de langage : 4 ne permet pas plus de 2 mots !! il doit croître de manière proportionnelle au nombre d'observations pour ne tuer que les chemins aberrants.