

PRÉSENTATION DE PROJET EDTS - RECONNAISSANCE DE PAROLE EN UTILISANT HMM BASÉ SUR CMU SPHINX

15 décembre 2015

Zhaolun Wang et Zenan Xu

Institut National des Sciences Appliquées



Sommaire



- 1 Introduction
- 2 Modèles acoustiques
- 3 Modèles de mots
- 4 Réalisation du projet
- 5 Amélioration
- 6 Conclusion

Introduction

Objectif



- Reconnaissance de parole en utilisant HMM basé sur CMU Sphinx

Les Modèles de Markov Cachés ou Hidden Markov Model



Définition

Un modèle de Markov caché (MMC) est un modèle statistique permettant de représenter un processus de Markov dont l'état est non observable.

Processus markov(Chaîne de Markov)

En mathématiques, un processus de Markov est un processus stochastique possédant la propriété de Markov. Dans un tel processus, la prédiction du futur à partir du présent n'est pas rendue plus précise par des éléments d'information concernant le passé.

Applications de HMM



- Reconnaissance de la parole.
- Traitement automatique du langage naturel.
- Reconnaissance de l'écriture manuscrite.
- Bio-informatique, notamment pour la prédiction de gènes.

Un exemple illustratif de HMM

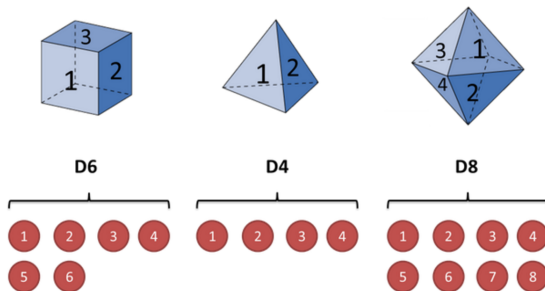


Figure 1 : Un exemple de HMM

Un exemple illustratif de HMM

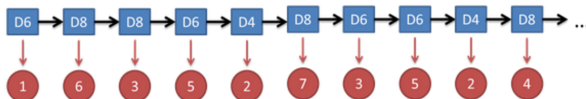


Figure 2 : Un exemple de HMM

- Les états observés : 1 6 3 5 2 7 3 5 2 4
- Les états cachés : D6 D8 D8 D6 D4 D8 D6 D4 D8

Algorithm de Viterbi



- L'algorithme de Viterbi permet de calculer la valeur la plus probable des états cachés du processus étant donnée des données observables.

L'Outil CMU Sphinx



Les CMU Sphinx comprend entre autres les outils suivants :

- Sphinx 2 : est un système de reconnaissance de la parole à grande vitesse. Il est habituellement employé dans des systèmes de dialogue et des système d'étude de prononciation.
- Sphinx 3 : est un système de reconnaissance de la parole légèrement plus lent mais plus précis.
- Sphinx 4 : Une réécriture complète du Sphinx en Java. Il offre à la fois la précision et la rapidité.
- Sphinxtrain : Une suite d'outils qui permet de créer le modèle acoustique .
- CMU-Cambridge Language Modeling Toolkit : Une suite d'outils qui permet de créer le modèle de langage.
- Sphinx Knowledge Base Tool : Un outil qui permet de créer le modèle de mots qui adapte son modèle de langage.

Les Modèles de Markov Cachés en speech to text



Nous considérons un signal acoustique S , le principe de la reconnaissance peut être expliqué comme le calcul de la probabilité $P(W|S)$ qu'une suite de mots (ou phrase) W correspond au signal acoustique S , et de déterminer la suite de mots qui peut maximiser cette probabilité. En utilisant la formule de Bayes, $P(W|S)$ peut s'écrire :

$$P(W|S) = P(W).P(S|W)/P(S)$$

- $P(W)$ est la probabilité a priori de la suite de mots W
- $P(S|W)$ est la probabilité de signal acoustique S , étant donné la suite de mots W
- $P(S)$ est la probabilité du signal acoustique
- $P(S|W)$ est nommé Modèle Acoustique, et $P(W)$ est nommé Modèle de Langage

Principe de la reconnaissance de la parole

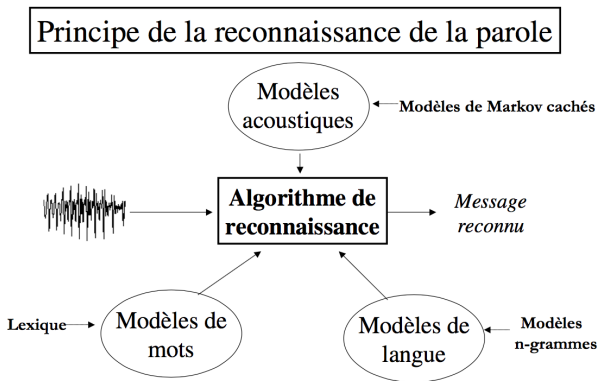


Figure 3 : Principe de la reconnaissance de la parole

Modèles acoustiques

Traitement acoustique : extraction de paramètres



- MFCC(Mel Frequency Cepstral Coefficients)
- LPCC(Linear Predictive Cepstral Coefficients)
- PLP(Perceptuqal Linear Predictive analysis)

Décodage acoustique et apprentissage



- Chaînes et modèles de Markov cachés
- Critère du maximum de vraisemblance
- Critère de Viterbi

Adaptation



- Méthode MLLR
- Méthode MAP

Création du Modèle acoustique en utilisant Sphinxtrain



Les données d'entrées sont composés, entre autre :

- d'un ensemble de fichiers acoustiques(corpus).
- d'un fichier de transcription qui contient l'ensemble de mots prononcés pour chaque enregistrement(fichier acoustique).
- d'un fichier qui définit la liste des phonèmes utilisées.

Modèle des N-Grammes



$$P(W_{1\dots k}) = P(W_1).P(W_2)|P(W_1). \dots .P(W_{N-1})|P(W_{1\dots N-2}). \prod_{l=N}^k P(W_l|W_{l-(N-1)\dots l-1})$$

Création de Modèle de langage en utilisant CMUCLMTK



La création d'un Modèle de langage statistique peut se résumer en trois étapes :

- Collecter des textes
- Transformer les textes en corpus
- Transformer le corpus en une distribution de probabilités

Modèles de mots

La dictionnaire phonétique



abbey aa bb ei
 abbot aa bb oo tt
 abbott aa bb au tt
 abbé aa bb ei
 abc aa bb ei ss ei
 abchac aa bb ch aa kk
 abcès aa bb ss ai
 abcès(2) aa bb ss ai zz
 abdallah aa bb dd aa ll aa
 abdel aa bb dd ai ll
 abdelati aa bb dd ei ll aa aa tt ii
 abdelatif aa bb dd ei ll aa tt ii ff
 abdelaziz aa bb dd ai ll aa zz ii zz
 abdelaziz(2) aa bb dd ei ll aa zz ii zz
 abdelazize aa bb dd ei ll aa zz ii zz
 abdelghani aa bb dd ei ll gg aa nn ii
 abdelhadi aa bb dd ei ll aa dd ii
 abdelhafid aa bb dd ei ll aa ff ii dd
 abdelhalim aa bb dd ei ll aa ll ii mm
 abdelhamid aa bb dd ai ll aa mm ii dd
 abdelhamid(2) aa bb dd ei ll aa mm ii dd
 abdelkader aa bb dd ai ll kk aa dd ai rr
 abdelkebir aa bb dd ei ll kk ei bb ii rr
 abdelkrim aa bb dd ai ll kk rr ii mm
 abdellah aa bb dd ai ll aa

Génération de dictionnaire



Il existe plusieurs outils qui nous permettent d'étendre une dictionnaire existante ou de générer une nouvelle dictionnaire :

- Phonetisaurus/sequitur-g2p ;
- espeak for C ;
- FreeTTS/OpenMary Java TTS ;
- etc.

Réalisation du projet

Analyser de résultat



Plusieurs raisons les plus possibles qu'on n'a pas très bien réussi :

- Testeur ;
- Modèle et dictionnaire ;
- Temps de recording ;
- etc.

Amélioration

Création de Modèle de langage en utilisant CMUCLMTK



La création d'un Modèle de langage statistique peut se résumer en trois étapes :

- Collecter des textes
- Transformer les textes en corpus
- Transformer le corpus en une distribution de probabilités

Conclusion