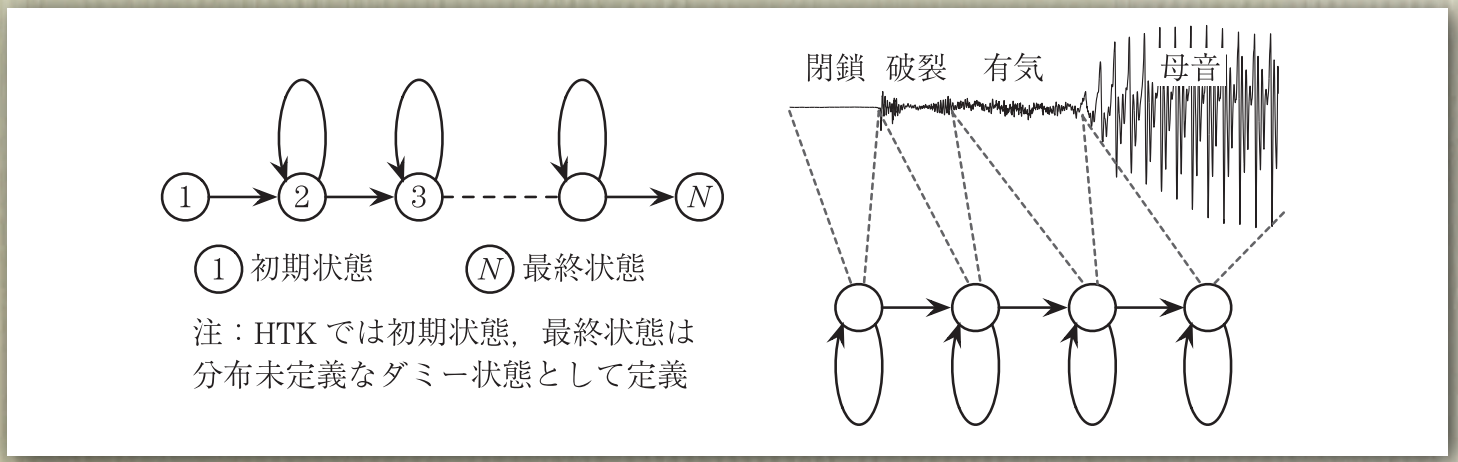
GMM-HMMからDNN-HMMへ

GMM-HMMとは？

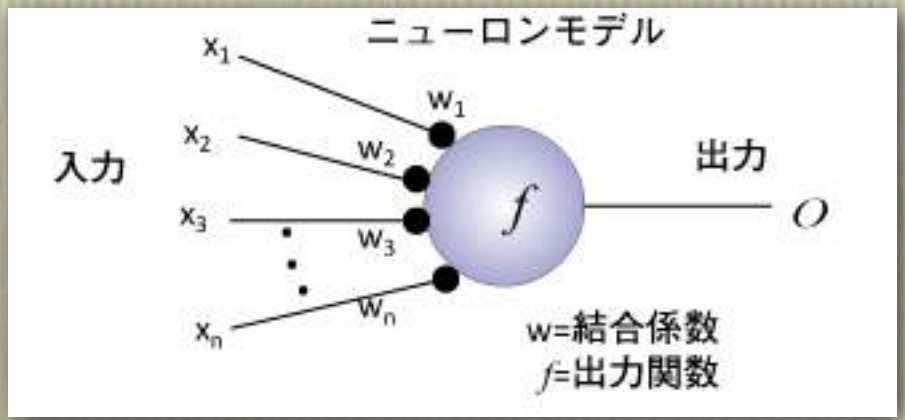
- 隠れマルコフモデル
- 時系列データの統計モデル
- 個々のデータを生成する情報源（分布＝状態）を考え，（音声認識の場合は）left-to-right の状態列（＝分布列）を想定する
- モデルを固定すると（与えられると），そのモデルからどのような時系列データが生成されやすいのか，の確率を与える
  - P(O | M=m)，分布としては Gaussian Mixture Model を使う
  - ある o に対する確率を計算できる P(O=o | M=m)

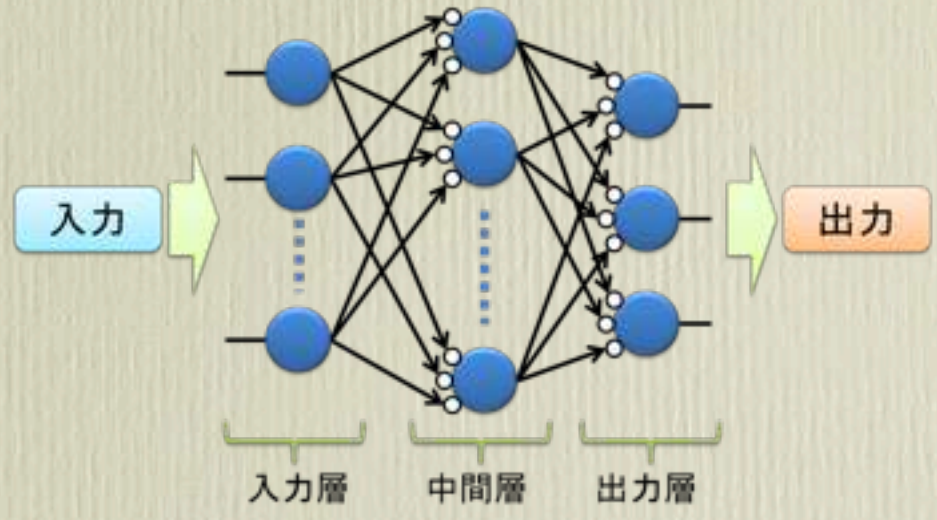


GMM-HMMからDNN-HMMへ

NN（Neural Network）とは？

- 正確にはANN（=Artificial Neural Network）
- ニューロンの結合パターンとそれによる情報伝搬のモデル
- ニューロンモデル
  - 入力ベクトルの各次元に重みを掛け，バイアス項を足しあわせたものを（非線形）関数 f を通して出力とする。$o = f(o') = f(\vec{w} \cdot \vec{x} + b)$
  - ある層の出力をベクトルとして考えると，重みを掛ける演算は行列演算となる。$\vec{o'}_{new} = W\vec{o}_{old} + \vec{b}$ その後 $\vec{o'}_{new}$ の各要素を f に通す。





https://thinkit.co.jp/article/30/2/ http://www.lab.kochi-tech.ac.jp/future/1110/okasaka/neural.htm

GMM-HMMからDNN-HMMへ

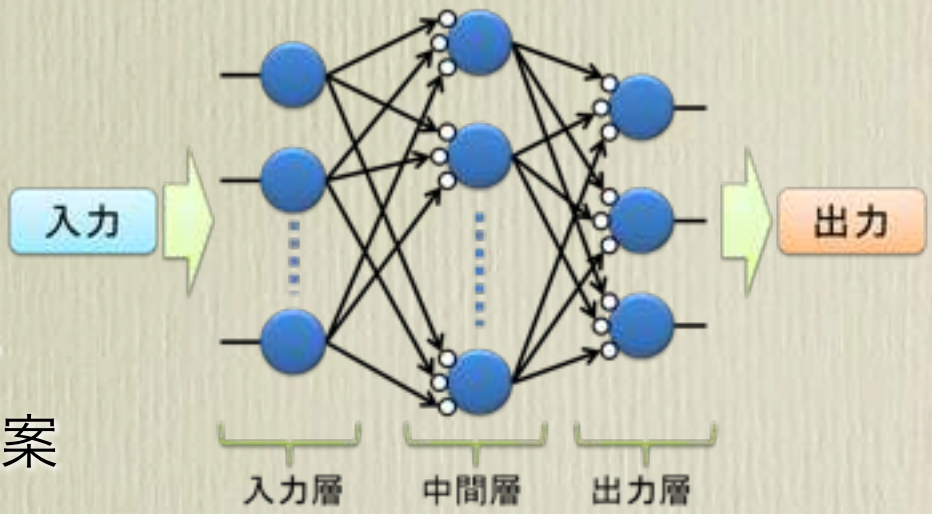
NNを分類器として使う，とは？

- 特徴ベクトルを入力として，出力はクラス事後確率とするNN
  - 文字認識，数字認識，音素認識，などなど
  - NNへの入力 $\vec{o}$ に対して出力が $P(c_i|\vec{o})$ となるよう，NNを学習
  - 出力が離散確率分布になるように学習する，とは？
    - 最終層の関数 f として，softmax 関数を使うことが多い。

$$P(c_i|\vec{o'}) = \frac{\exp(o'_i)}{\sum_i \exp(o'_i)}$$

NNに対してDNNとは？

- 中間層の数を増やしたもの
  - 最近では数十層という実装もある
  - 従来は総数を増やすと学習が困難であったが，それを解決する方法が提案



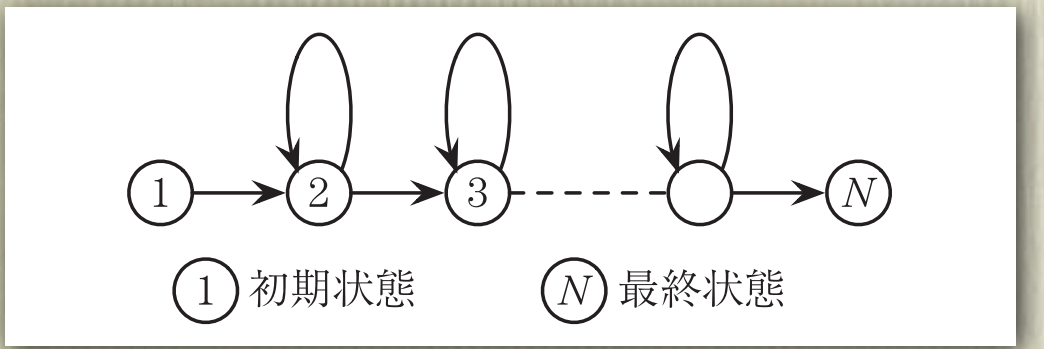
GMM-HMMからDNN-HMMへ

音声特徴をDNNに入れるとどうなる？

- 音素事後確率の推定器として機能
  - $\vec{x} \longrightarrow P(c_i|\vec{x})$
- 音素 HMM は全種類の音素に対して，数千の状態を用意し，
  個々の音素HMMは，それらの状態を「共有」することが多い。
  - 状態共有 HMM（音素分類のためのトップダウンクラスタリング）
- 結局，音素事後確率は，音素状態事後確率に $\vec{x} \longrightarrow P(c_i|\vec{x})$
  - 入力特徴が数千次元のベクトルに

再度，HMMに戻ります

- GMM-HMM の音声認識では $P(\vec{x}|S_i)$ を計算した。



GMM-HMMからDNN-HMMへ

ベイズの定理でひっくり返します。

- GMM-HMM の音声認識では $P(\vec{x}|S_i)$ を計算した。
  - 各状態＝GMM
- ひっくり返します。

$$P(\vec{x}|S_i) = \frac{P(\vec{x}, S_i)}{P(S_i)} = \frac{P(S_i|\vec{x})P(\vec{x})}{P(S_i)}$$

  - ある $\vec{x}$ を識別する問題を考える時は, $P(\vec{x})$ は定数として扱える。
  - $P(S_i)$ は，音声データに状態 S_i 相当の音はどの程度の頻度で出現するのか，に相当する。コーパスを使って事前に求めておく。
- 結局, $P(\vec{x}|S_i)$ を求めるために
  - GMM を使ってガウス分布の確率値として求めるのではなく,
  - DNN の事後確率を使って，間接的に（遠回りして）求める。
  - これを，DNN-HMM と言う。
- なんで，こんな遠回しの方法が良いのか？

GMM-HMMからDNN-HMMへ

考えられる理由

- 音声特徴量は本当にガウス分布（の混合分布）に従うのか？
  - ガウス分布に従わないなら，何らかのミスマッチが起こる。
  - DNN の場合，特徴量分布を陽に仮定していない。
- CEP ではなく，スペクトルがそのまま用いられることが多い。
  - DNN は基本的に，行列演算＋非線形関数
  - FFTは入力ベクトルに対する一次変換
  - であれば，スペクトルを入れてDNNを学習すれば，FFTっぽい行列が第一層目として学習されるはず・・・？
    - そもそもCEPって最適な特徴量なの？
    - より raw 特徴量を入力して，あとは DNN に任せた方がよい？
- 数フレーム分のスペクトル特徴が入力されることが多い。
  - DNN の場合，入力特徴量の次元を増やすことが比較的楽。
  - GMM の場合，より多くの学習データ量が必要となる。

GMM-HMMからDNN-HMMへ

GMM-HMMとDNN-HMM

両者の性能差

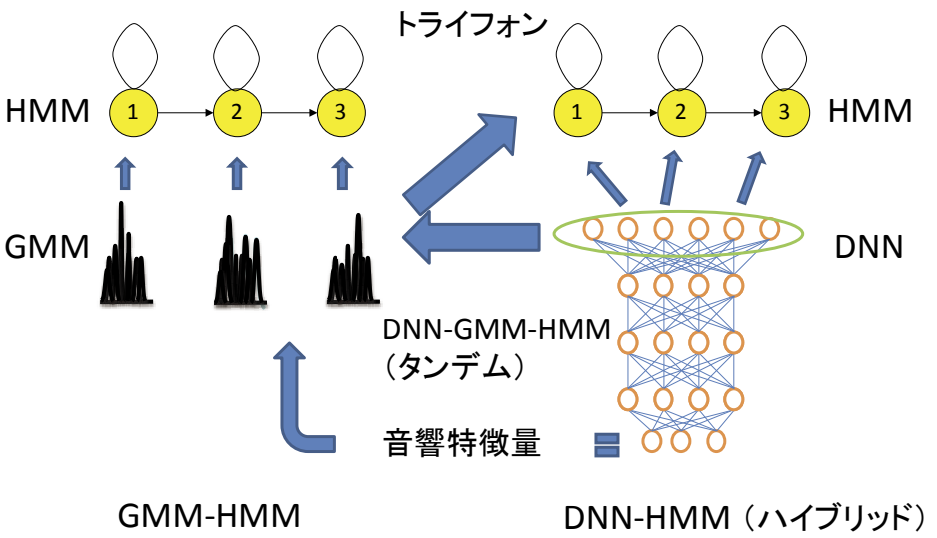


図２　GMM-HMM と DNN-HMM

表２　GMM-HMM と DNN-HMM の比較

	学習データ(時間)	GMM-HMM 単語誤り率	DNN-HMM 単語誤り率
TIMIT 音素認識	10	27.3%	22.4%
Switchboard 電話音声	300	23.6%	17.1%
Google 音声検索	5870	16.0%	12.3%
JNAS 日本語新聞記事	85	6.8%	3.8%
CSJ 日本語講演音声	257	20.0%	16.9%

河原達也「音声認識技術の展開」PRMU2015-111(2015-12)より