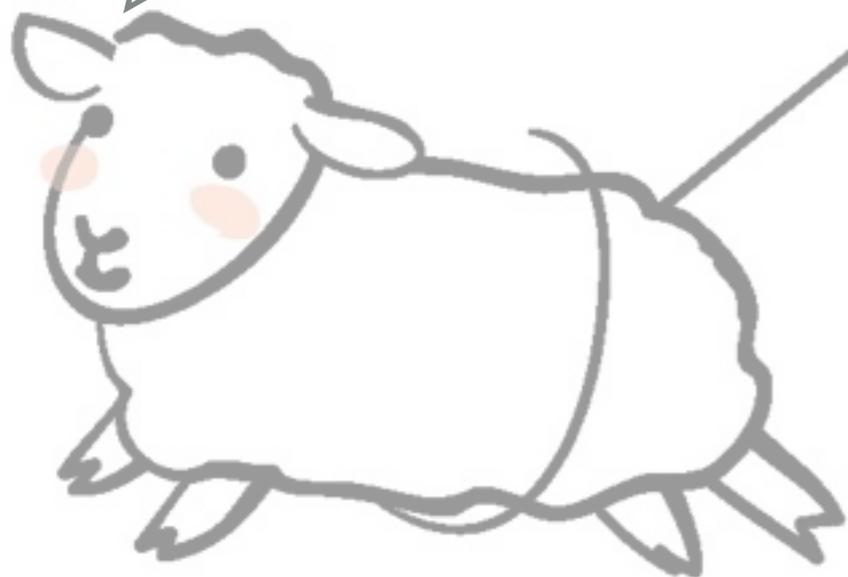


自動音声認識システム JULIUS の使い方

これで君も音声
認識マスター！

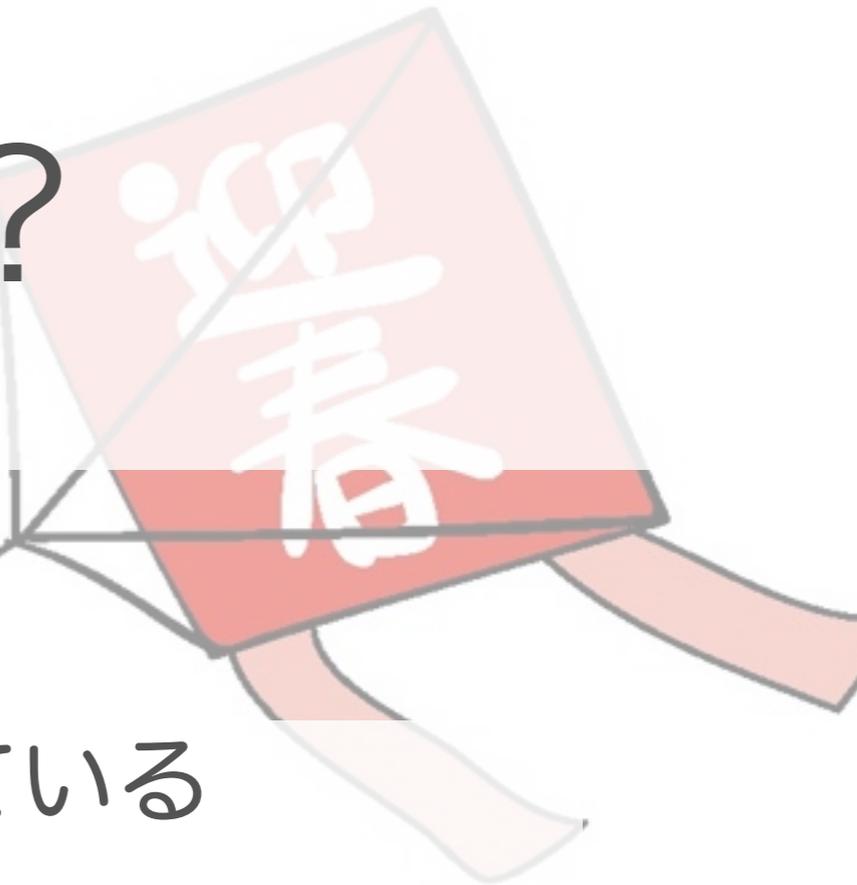


JULIUS とは？

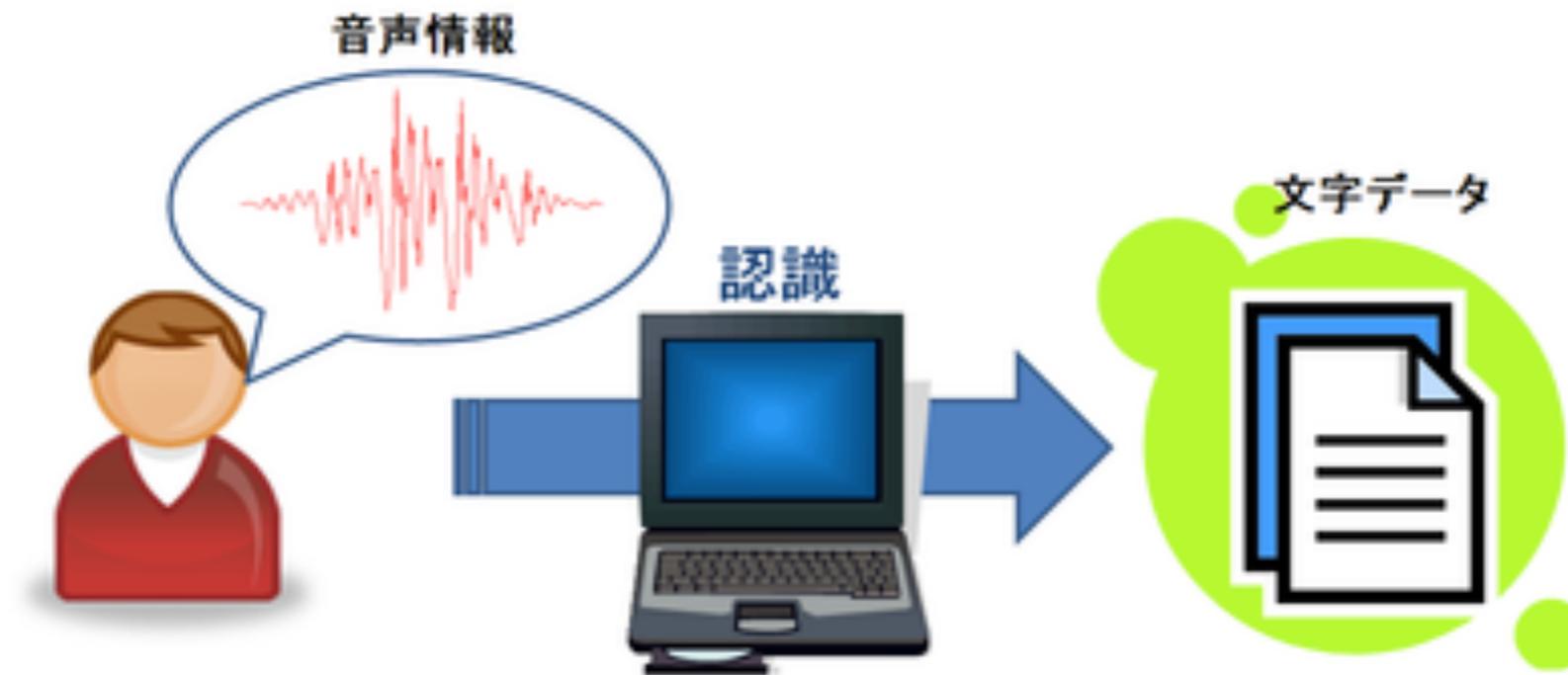


- ▶ 名古屋工業大学のチームが開発している音声認識エンジン
 - ▶ オープンソース
 - ▶ <http://julius.sourceforge.jp/>

▶ 今回は認識部分をこれをベースに構築



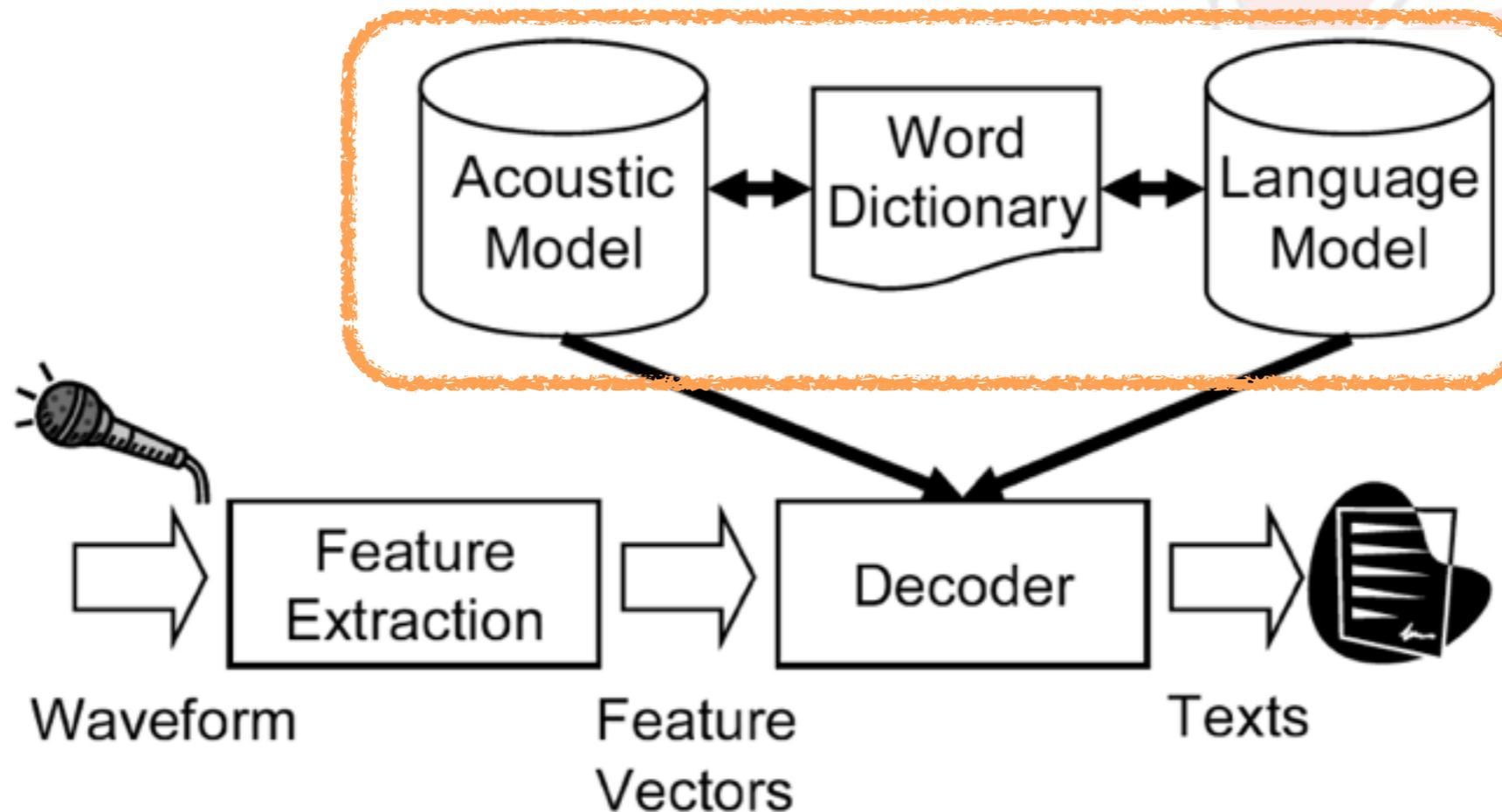
自動音声認識とは？



- ▶ 孤立単語認識：区切って発声した単語を認識
 - ▶ タスクを想定した少数の語彙であることが多い
- ▶ **連続音声認識**：連続して発声した文を認識
 - ▶ 多数の語彙を想定し，言語情報の利用が重要



統計的手法による音声認識



▶ 音響モデル

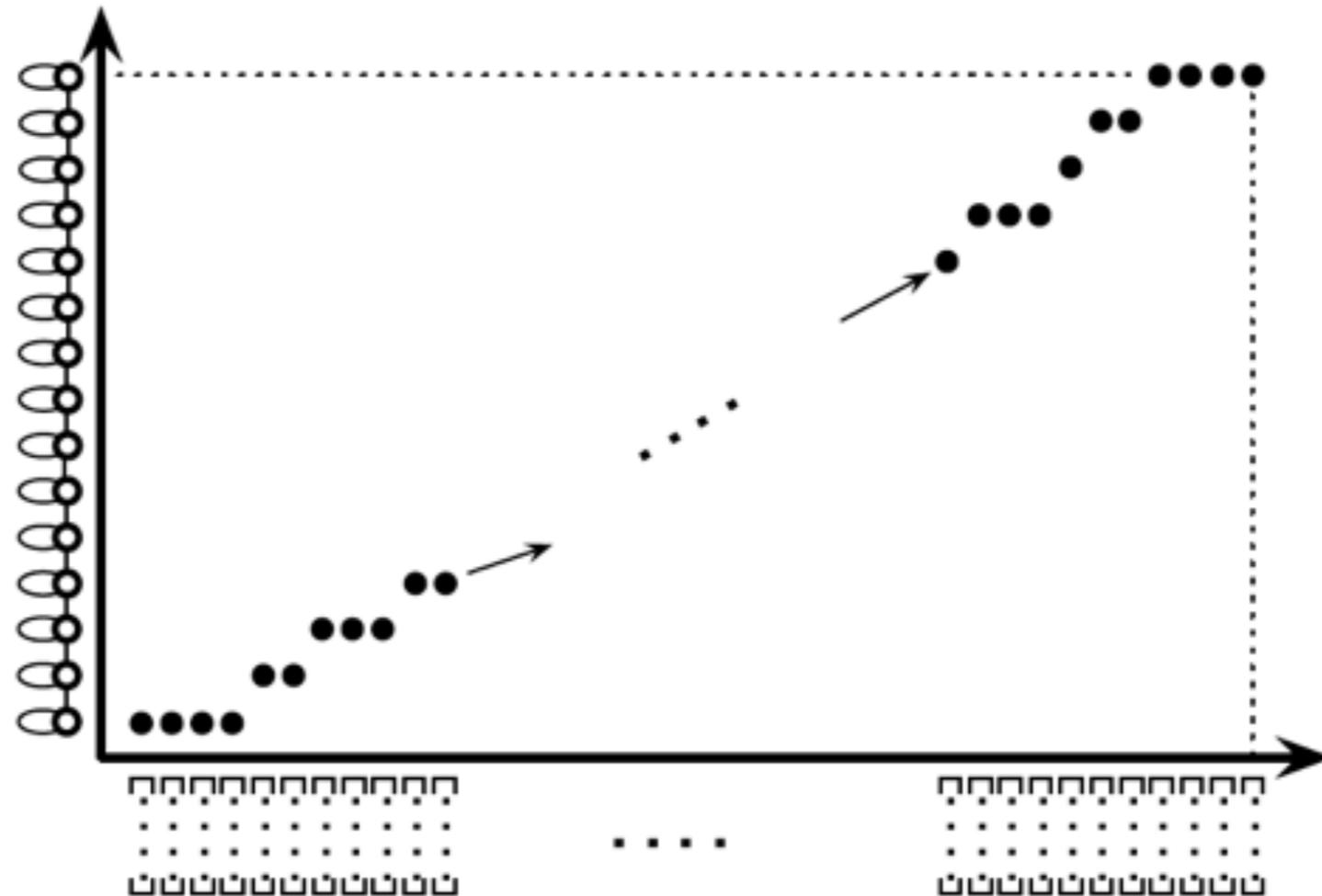
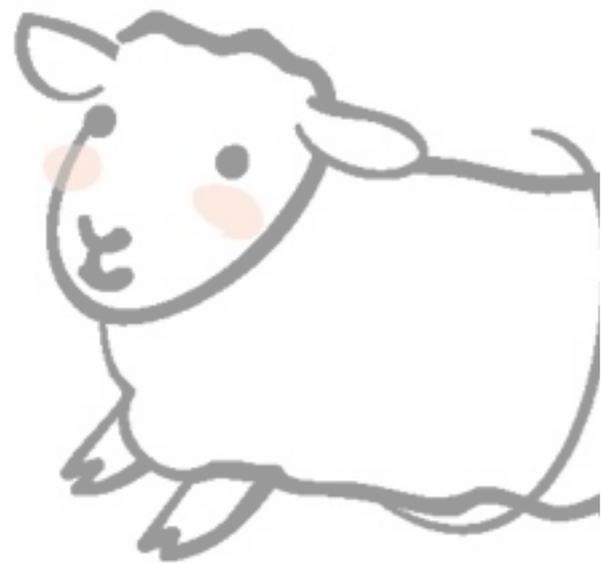
- ▶ 音響特徴量の分布のモデル (Hidden Markov Model など)

▶ 言語モデル

- ▶ 言語情報のモデル (N-gram、文法ネットワーク など)

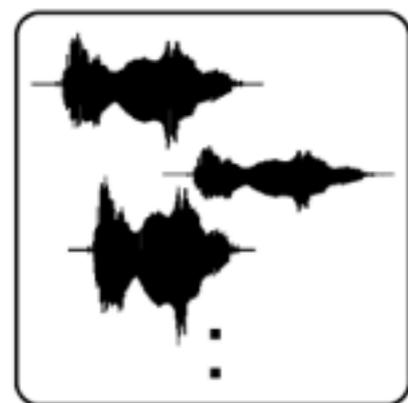
デコーダ

- ▶ ネットワークにおける最尤のパスを計算
 - ▶ 音声は時系列データなので経路探索が必要

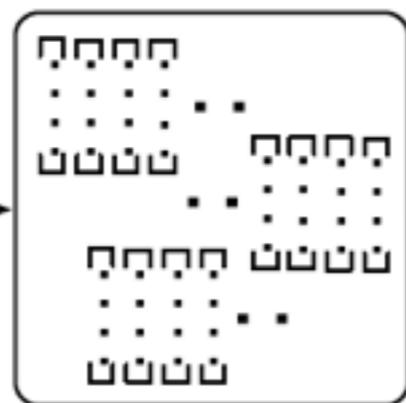


音響モデル

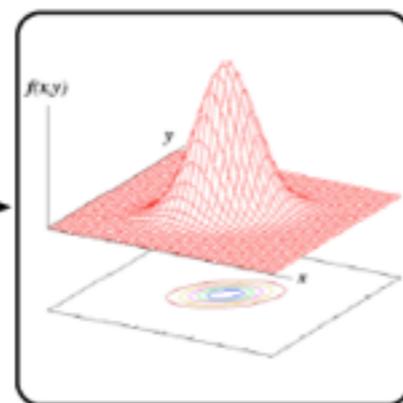
- ▶ モデル化する単位
 - ▶ monophone : 音素毎にモデル化
 - ▶ triphone : 3つ組の音素単位でモデル化 (調音結合を想定)
- ▶ Hidden Markov Model
 - ▶ 各状態が分布をもつ状態遷移モデル
 - ▶ 各状態はGaussian Mixture Model や Deep Neural Network 等でモデル化



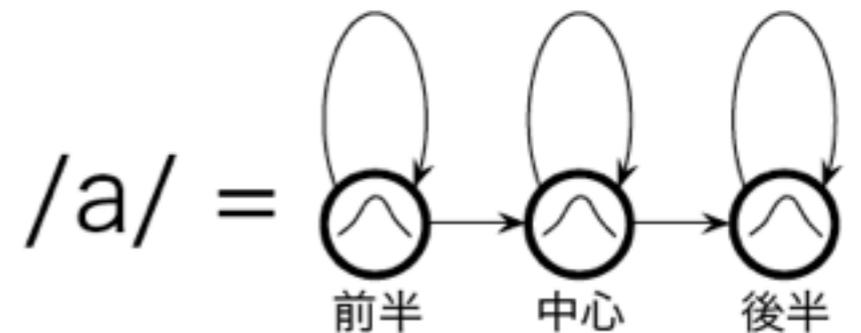
多数の/a/



ケプストラム



多次元正規分布



言語モデル

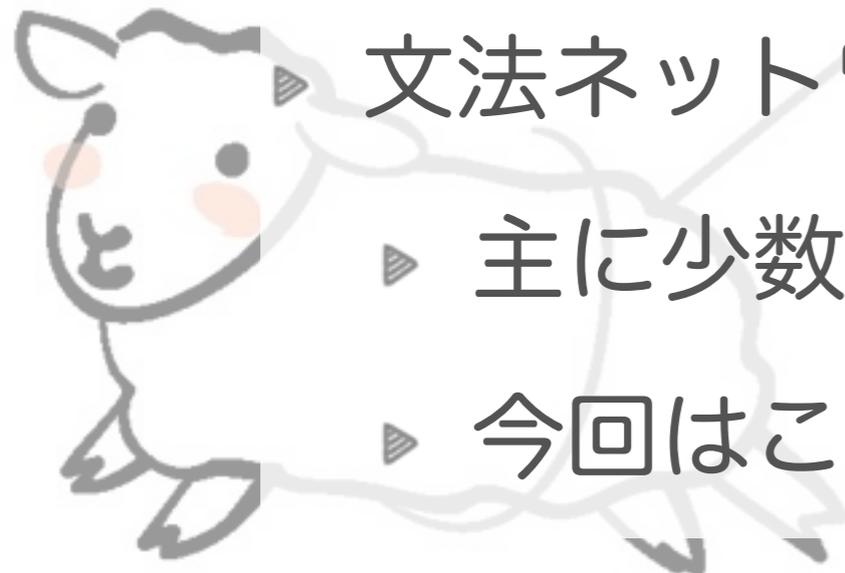
- ▶ N-gram

- ▶ 直前のN-1単語から次の単語の確率を計算
- ▶ 主に大語彙音声認識に利用

$$P(w_1, \dots, w_n) = \sum_{i=1}^n P(w_i | w_{i-1}, \dots, w_{i-N+1})$$

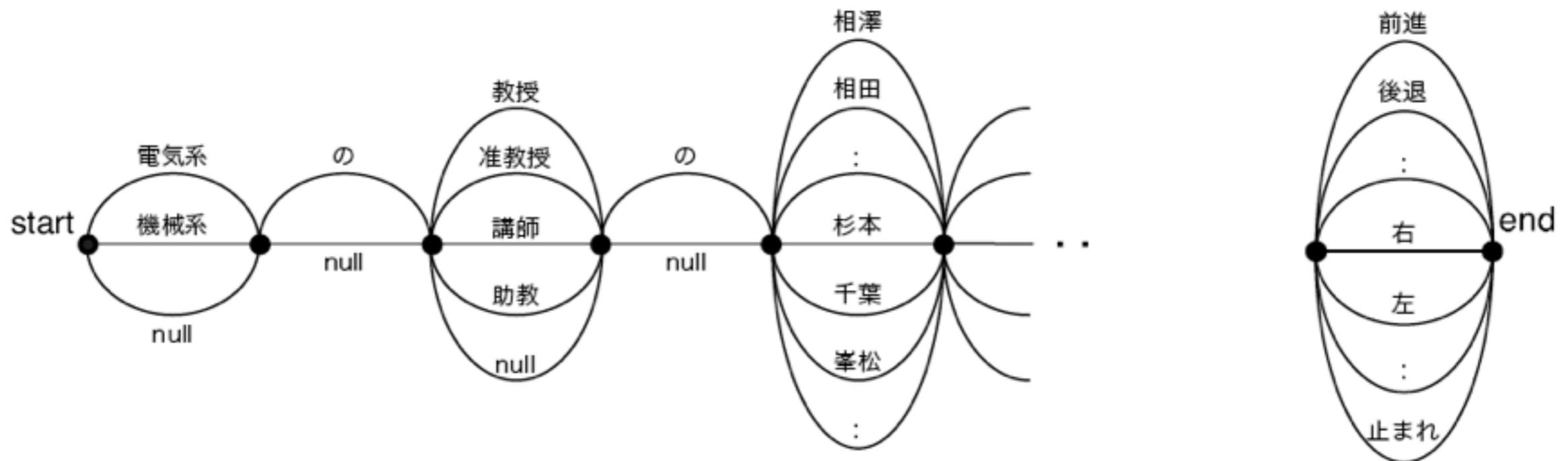
文法ネットワーク

- ▶ 主に少数の語彙による定形パターンの認識
- ▶ 今回はこちらを利用



ネットワーク文法

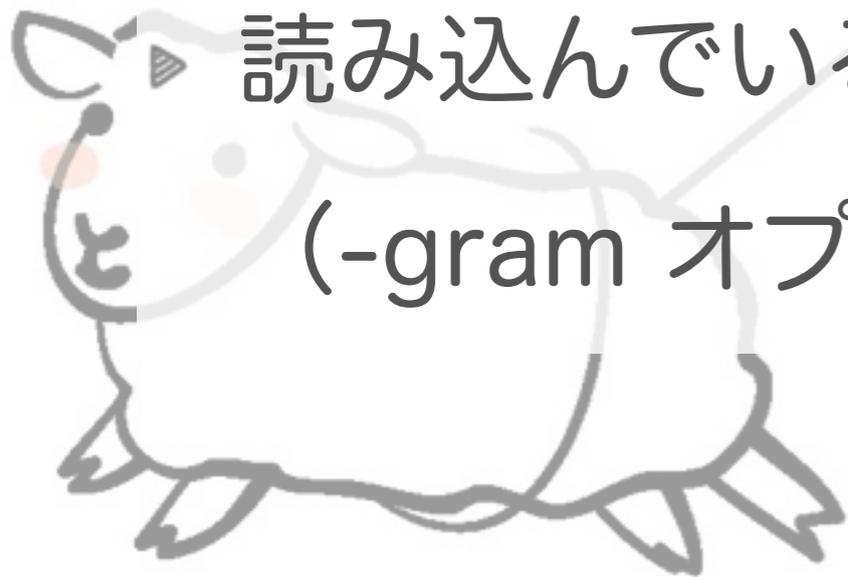
- ▶ すべての単語を想定すると高コスト
- ▶ あらかじめ入力を絞ったオートマトンを構築する
 - ▶ 高速に認識可能
 - ▶ 誤りの軽減



音声認識の実行方法

- ▶ マイクをつないで Julius を実行！
 - ▶ `julius -C grammar-mic.jconf`
 - ▶ `grammar-mic.jconf` : 設定ファイル

- ▶ 読み込んでいる文法ファイルを編集すべし！
(`-gram` オプション)



文法と単語の編集

- ▶ 単語の追加・削除
 - ▶ .yomi ファイルの作成

% はクラスのラベル！

```
% NAME
小泉 こいずみ
片山 かたやま
森山 もりやま
田中 たなか
塩川 しおかわ
遠山 とーやま
坂口 さかぐち

% FILL_E
お願い おねがい
お願いします おねがいします
さん さん
さんお願い さんおねがい
さんお願いします さんおねがいします

% FILL_B
えー えー
えーと えーと
えーと えーっと
あの一 あの一

% ANSWER
ええ えー
はい はい
いいえ いいえ
.
.
.
```



文法と単語の編集

- ▶ 単語の追加・削除
 - ▶ .yomi ファイルの作成
 - ▶ .voca ファイルへ変換
 - ▶ yomi2voca を使用

% はクラスのラベル！

```
% NAME
小泉 k o i z u m i
片山 k a t a y a m a
森山 m o r i y a m a
田中 t a n a k a
塩川 s h i o k a w a
遠山 t o : y a m a
坂口 s a k a g u c h i

% FILL_E
お願い o n e g a i
お願いします o n e g a i s h i m a s u
さん s a N
さんお願い s a N o n e g a i
さんお願いします s a N o n e g a i s h i m a s u

% FILL_B
えーe:
えーと e: t o
えーと e: q t o
あの一 a n o:

% ANSWER
ええ e:
はい h a i
いいえ i i e
.
.
.
```



文法と単語の編集

- ▶ 単語の追加・削除
 - ▶ .yomi ファイルの作成
 - ▶ .voca ファイルへ変換
 - ▶ yomi2voca を使用
- ▶ 文法の変更
 - ▶ .grammar ファイルの編集

```
S : NS_B NAME NS_E
S : NS_B NAME FILL_E NS_E
S : NS_B FILL_B NOISE NAME FILL_E NS_E
S : NS_B FILL_B NS_E
S : NS_B ANSWER NS_E
```

ルールも組めるよ！

文脈自由文法

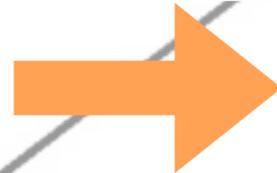


ループの作成例

▶ nishino.yomi

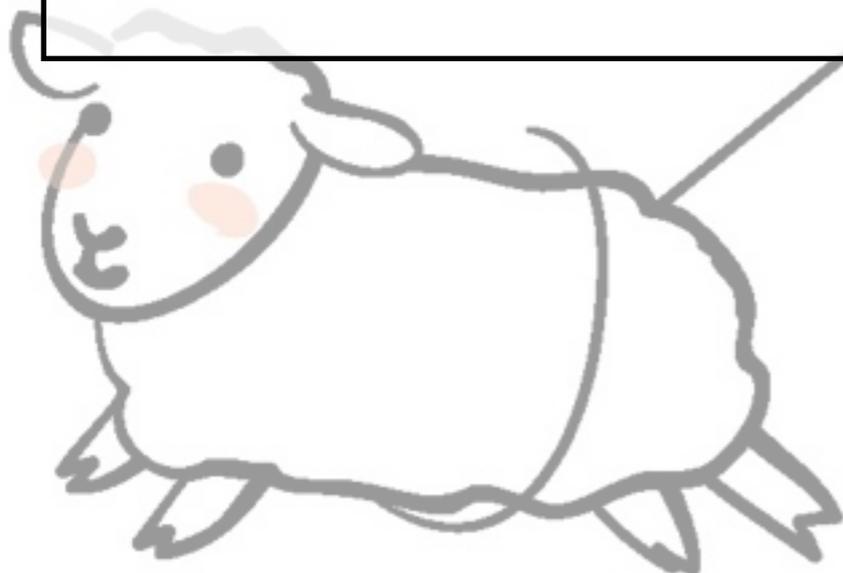
```
% AITAKUTE  
会いたくて      あいたくて  
  
% FURUERU  
震える   ふるえる  
  
## 無音用エントリ  
% NS_B          # 文頭無音  
silB           silB  
% NS_E          # 文末無音  
sile          sile
```

yomi2voca



▶ nishino.voca

```
% AITAKUTE  
会いたくて      a i t a k u t e  
  
% FURUERU  
震える   f u r u e r u  
  
## 無音用エントリ  
% NS_B          # 文頭無音  
silB           silB  
% NS_E          # 文末無音  
sile          sile
```



ループの作成例

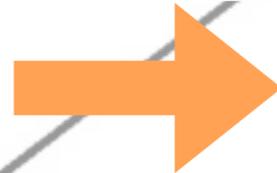
▶ nishino.yomi

```
% AITAKUTE
会いたくて      あいたくて

% FURUERU
震える   ふるえる

## 無音用エントリ
% NS_B          # 文頭無音
silB           silB
% NS_E          # 文末無音
silE           silE
```

yomi2voca



▶ nishino.voca

```
% AITAKUTE
会いたくて      a i t a k u t e

% FURUERU
震える   f u r u e r u

## 無音用エントリ
% NS_B          # 文頭無音
silB           silB
% NS_E          # 文末無音
silE           silE
```

▶ nishino.grammar

```
S          : NS_B AITAKUTE_LOOP FURUERU NS_E
AITAKUTE_LOOP : AITAKUTE_LOOP AITAKUTE
AITAKUTE_LOOP : AITAKUTE
```



ループの作成例

- ▶ generate で結果のチェック！

```
Stat: init_voca: read 4 words
Reading in term file (optional)... done
4 categories, 4 words
DFA has 5 nodes and 5 arcs
```

```
-----
silB 会いたくて 会いたくて 会いたくて 震える silE
silB 会いたくて 震える silE
silB 会いたくて 会いたくて 震える silE
silB 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 震える silE
silB 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 会
silB 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 震える silE
silB 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 震
silB 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 会いたくて 会
```

文法と単語の編集

- ▶ 単語の追加・削除
 - ▶ .yomi ファイルの作成
 - ▶ .voca ファイルへ変換
 - ▶ yomi2voca を使用
- ▶ 文法の変更
 - ▶ .grammar ファイルの編集
- ▶ ネットワークのコンパイル
 - ▶ mkdfa を使用
 - ▶ .dfa .dict .term を生成してくれる
 - ▶ generate でサンプルの確認が可能





色々な細かいパラメータ調整ができます

詳細はJuliusbookを読んでね！

<http://julius.sourceforge.jp/>

