

# ROBOMECH 2007 in Akita

## RTミドルウェアによる 学習・推論コンポーネント

---

2007年5月11日

玄葉 誠

株式会社アドイン研究所

# RTミドルウェアによる学習・推論コンポーネント

The learn and reason component on RT-Middleware

玄葉 誠 (アドイン研)

Makoto GENBA, AdIn Research Inc

中村 健 (アドイン研)

Ken Nakamura, AdIn Research Inc

安藤 慶昭 (産総研)

Noriaki ANDO, National Institute of Advanced Industrial  
Science and Technology

神徳 徹雄 (産総研)

Tetsuo KOTOKU, National Institute of Advanced Industrial  
Science and Technology

# 1. はじめに

本発表では RTミドルウェア を用いて開発した 学習・推論コンポーネント について述べる。

具体的には、

RTミドルウェア

学習・推論

コンポーネント化作業

デモシステム

学習・推論コンポーネントの利用

について説明する。



## 2. RTミドルウェア (1)

### RTミドルウェアとは？

モジュールの組合せによりロボットシステムを構築する為のプラットフォーム

### そのメリットは？

コンポーネント再利用による工数削減  
ネットワークに透過なシステムの実現  
言語 / OSに非依存な環境

### 誰が決める？

RTコンポーネント標準仕様が 2006年9月にソフトウェア標準化団体(OMG)に採択

## 2. RTミドルウェア (2)

### 何が提供されるの？

コンポーネントフレームワーク

標準的に再利用されるソフトウェア部品群

ライブラリ群

標準サービス群

### OpenRTM-aist-0.4.0 とは？

産総研・知能システム研究部門が開発した、  
RTコンポーネント標準仕様に基づくプロトタイプ

# 3. 学習・推論 (1)

## 学習とは？

サンプルデータ集合を対象に解析を行い、  
有用な規則やルール( = **知識**)を抽出すること  
(人工知能研究の一分野である機械学習)

## 推論とは？

学習で得た**知識**をもとに、新たなデータから、  
新しい結論を得ること

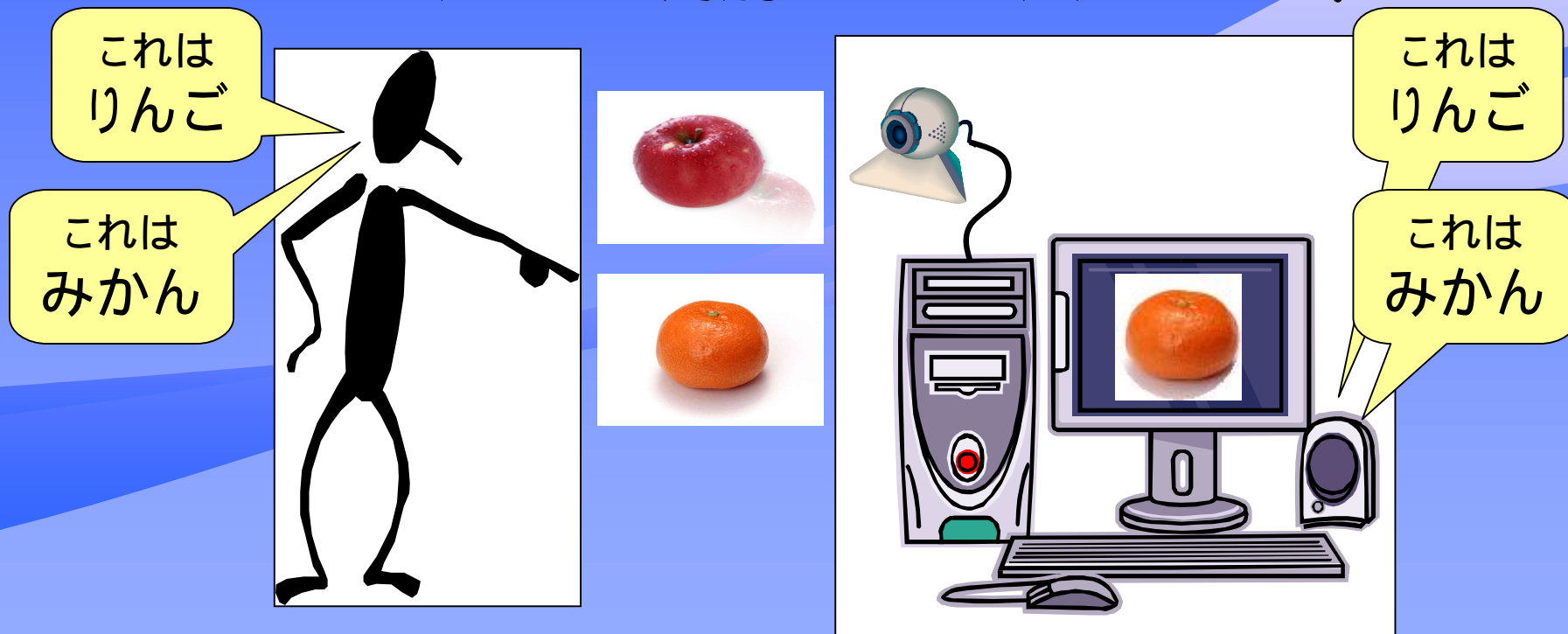
## -RNAとは？

株式会社アドイン研究所が開発した、  
ファジィ・ニューロ**学習・推論エンジン**

### 3. 学習・推論 (2)

#### “りんご”と“みかん”の判別

りんごとみかんを判別できない人はいない。



それを機械にやらせるとすると...

# 3. 学習・推論 (3)

## プログラミングするなら

### 計測可能データを決める

赤の強度 [red], 緑の強度[green], 青の強度[blue]

フィレ径 (図形を包絡する直方体の辺の平均) [size]

縦横比[aspect]

### 判断基準を決める

“もし色が赤くて少し縦長の直径8cmの球形をしているならば、りんごである”

### If-then 形式で表現する

```
if ((red 8.0) and (green 2.0) and (blue 2.0)
    and (7.0 size 9.0) and (1.0 aspect 1.3))
then (出力 = りんご)
```

でも、If-then では全てを表現できていないし、  
対象によっては、このように決めれるとは限らない。



# 3. 学習・推論 (4)

## -RNAなら

### 計測可能データを定める

赤の強度 [red], 緑の強度[green], 青の強度[blue]

フィレ径 (図形を包絡する直方体の辺の平均) [size]

縦横比[aspect]

### サンプルデータを学習する

red=8.0, green=2.0, blue=2.0, size=9.0, aspect=1.3 → りんご

red=7.8, green=2.1, blue=1.9, size=8.5, aspect=1.2 → りんご

red=6.9, green=8.9, blue=2.1, size=6.2, aspect=1.8 → みかん

red=7.1, green=8.4, blue=2.5, size=5.9, aspect=1.6 → みかん

### 新たな観測データで推論する

red=7.9, green=2.2, blue=1.8, size=8.4, aspect=1.3

→ りんご (合致度:82%)

# 4. コンポーネント化作業 (1)

## 機能設計

### コンポーネントの粒度を考える

どこでコンポーネントを分離するかを決める  
データ取得・特徴量算出機能と学習・推論機能は別のコンポーネントとする。

### 実現する機能を決める

学習機能 (サンプルデータを与えて学習する機能)  
推論機能 (観測データで推論する機能)

### コンポーネントのインターフェースを決める

In/OutPort のデータ内容とデータ型を定義  
ServicePortなら引数や戻り値を定義 (idlを定義)

### アクティビティ状態に対応する処理を決める

onActivated関数, onExecute関数等で行なう処理を決める。

## 4. コンポーネント化作業 (2)

### rtc-templateによる雛型作成

```
#!/bin/bash
```

```
rtc -template -bcxx ¥
```

```
-module -name=BetaRnaPort -module -type='BetaRnaPort' ¥
```

```
-module -desc='BetaRna in/output component' ¥
```

```
-module -version=0.1 -module -category=Generic ¥
```

```
-module -comp -type=COMMUTATIVE -module -act -type=SPORADIC ¥
```

```
-module -max -inst=10 ¥
```

```
--inport=DataCount:TimedShort --inport=DataAry:TimedDoubleSeq ¥
```

```
--inport=ID:TimedShort --inport=ParamDir:TimedString ¥
```

```
-output=ResultID:TimedShort -output=Trusty:TimedDouble
```

## 4. コンポーネント化作業 (3)

### 処理実装

#### 関数をコメントインする

rtc-templateで作成した雛型では、アクティビティに対応した関数はコメントアウトされているので、必要な関数をコメントイン

- 学習・推論コンポーネントでは、  
onActivated関数、onExecute関数をコメントイン

#### 処理を実装する

コメントインした関数に、状態や引数に応じた処理を実装する

- 学習・推論コンポーネントでは、  
onActivated関数、onExecute関数で、対応する -RNA  
ライブラリの関数をCallする

# 4. コンポーネント化作業 (4)

## 学習・推論コンポーネント仕様 (Port版)

### 入力ポート (InPort) 仕様

ポート名称	データ型	説明
ID	TimedShort	本ポートで、動作モード (学習モード/推論モード) の切り替えを行う。学習モード時は、教示のために、サンプルデータのグループを指定する役割も併せ持つ。 (学習モード) グループID, (推論モード) -1
ParamDir	TimedString	データディレクトリ (学習パラメタファイル、学習結果ファイルを格納するディレクトリ) を指定する。なお、データディレクトリは、コンポーネントサーバに存在する必要がある。
DataCount	TimedShort	学習モード時はサンプルデータの、推論モード時は、判別対象データの、 <b>特徴量の項目数</b> を指定する。
DataAry	TimedDoubleSeq	学習モード時はサンプルデータの、推論モード時は、判別対象データの、 <b>特徴量の配列</b> を指定する。各特徴量は、0以上1以下でなければならない。

### 出力ポート (OutPort) 仕様

ポート名称	データ型	説明
Result	TimedShort	推論モード時は、推論結果のグループIDが出力される。 (学習モード時は、入力ポートIDで指定した、サンプルデータのグループIDが、そのまま出力される。)
Trusty	TimedDouble	推論モード時のみ合致度 (推論結果が信頼できる度合い、0以上1以下) を出力する。(学習モード時は、-1が出力される。)

## 4. コンポーネント化作業 (5)

### 学習・推論コンポーネント仕様 (Service版)

関数名称	戻り値	戻り値の内容	引数	引数の内容
Init	short	処理結果	string sRnaDir short iInputCh	パラメータディレクトリ 特徴量の個数
Exec	short	処理結果	short iTalkerID DArray(*) aInputData short& iResultID double& dTrusty	話者ID 特徴量データ配列 推論結果の話者ID 確信度

(\*) DArray : sequence<double>のこと。

# 5. デモシステム (1)

## 話者認識システム

音声为谁の声かを判断するシステム

学習

「おはようございます」(Aさん) x 3回

「おはようございます」(Bさん) x 3回

「おはようございます」(Cさん) x 3回

「おはようございます」(Dさん) x 3回

「おはようございます」(Eさん) x 3回

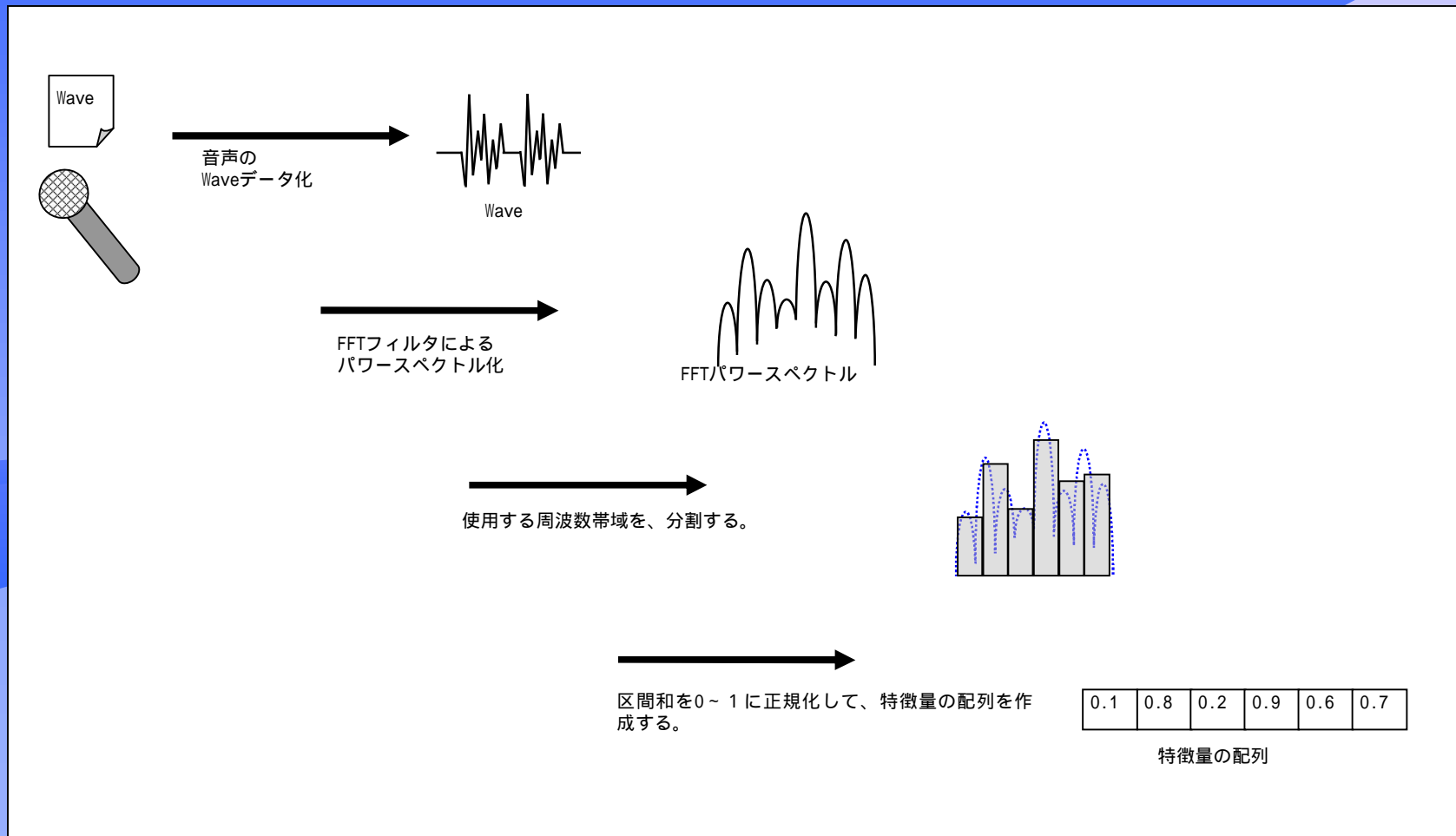
推論

「おはようございます」(?さん)

- > Cさん(合致度:95%)

# 5. デモシステム (2)

## 音声の特徴量化





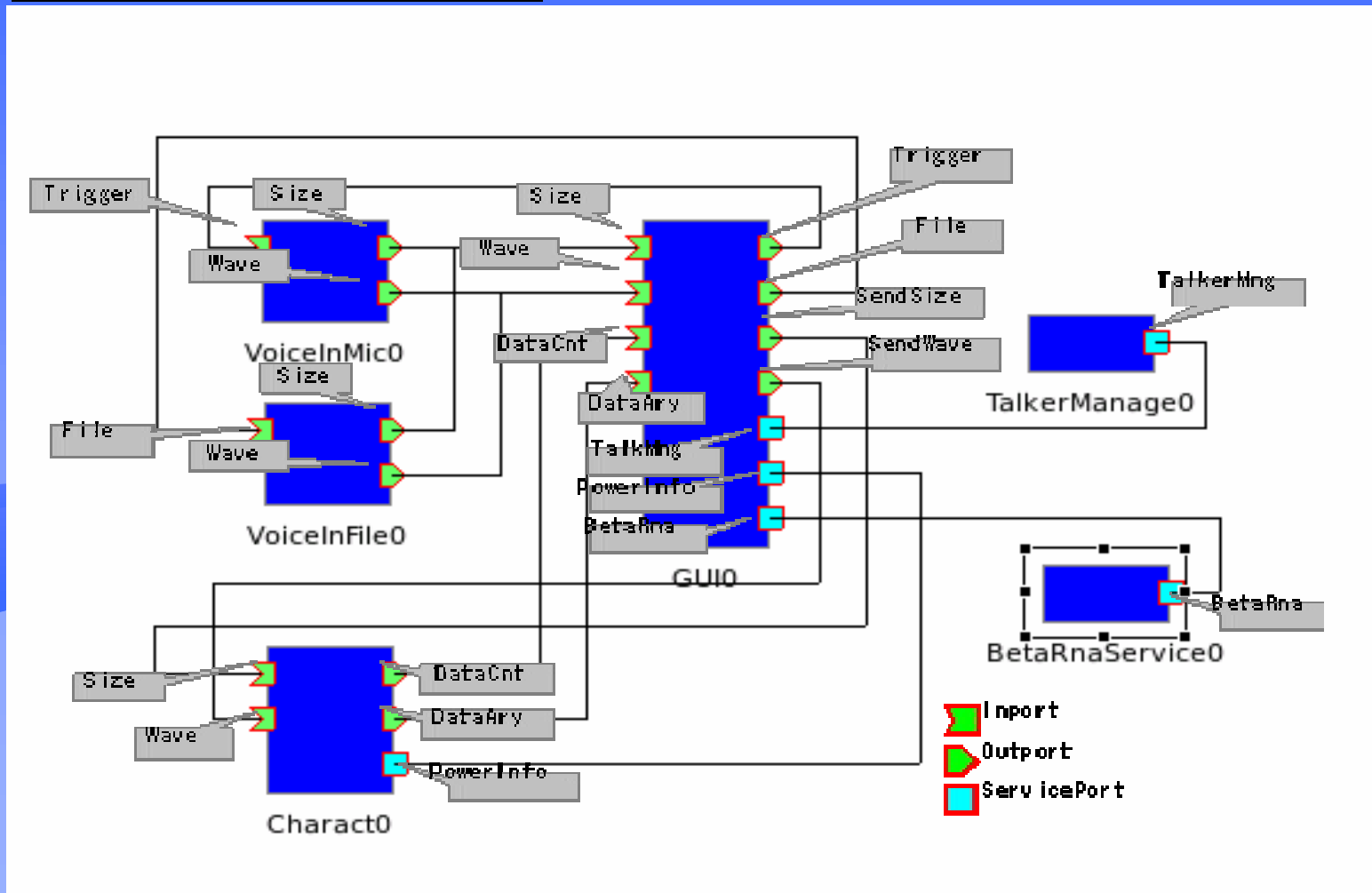
# 5. デモシステム (3)

## コンポーネント構成

コンポーネント種類	コンポーネント名	機能
音声入力 コンポーネント (マイク版)	VoiceInMic	入力開始トリガを受けて、マイクからの入力をWave形式で出力する。(内部で、音声入力ツール『Julius addtool』を使用している。)
音声入力 コンポーネント (ファイル版)	VoiceInFile	入力ファイル名を受けて、ファイルのデータをWave形式で出力する。
話者管理 コンポーネント	TalkerManage	話者名と識別ID(話者に割り振られるID)の関係を管理する。
音声特徴量抽出 コンポーネント	Charact	入力されたWaveデータをFFT解析し、パワースペクトルを求め、周波数帯別に平均した値(これを音声の特徴量とする)を出力する
学習・推論 コンポーネント	BetaRnaService	入力された音声の特徴量と、識別IDとの関係を学習する。入力された音声の特徴量が、どの識別IDに属するものか(=話者が誰か)を推論する。
ユーザインター フェース コンポーネント	GUI	ユーザに、話者認識システムのインターフェースを提供する。他のコンポーネントの、上位のコンポーネントとして働く

# 5. デモシステム (4)

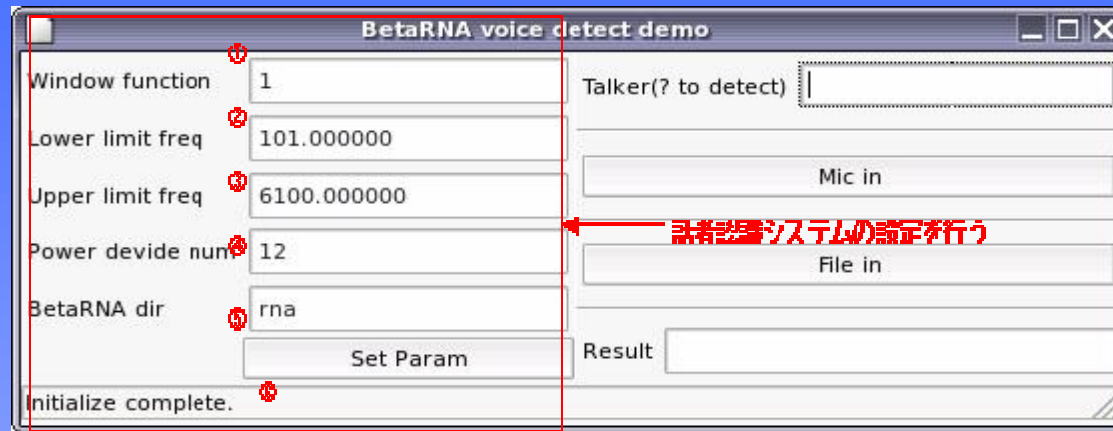
## RtcLinkでの接続



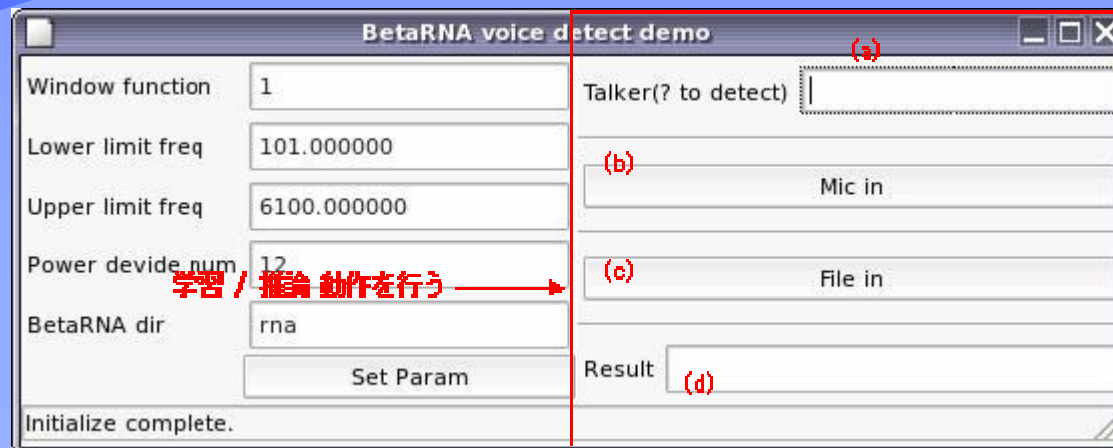
# 5. デモシステム (5)

## GUIでの操作

設定



学習  
推論



# 6. 学習・推論コンポーネントの利用

## 学習・推論コンポーネントとは？

数値データによる識別器

## 適する対象とは？

数値データが得られる

サンプルデータが得られる

識別結果が明確である(教示できる)

## 具体的には

物体識別(画像による物の認識)

状況判断(様々な入力数値から状況を判断)

Etc...

→ プログラムで書けない判断機能

## 7. おわりに

- 本発表では RTミドルウェア を用いて開発した、学習・推論コンポーネント について述べた。
- RTミドルウェア が普及し相互に接続できる RTコンポーネント が増えていけば、ロボットシステムの開発効率は飛躍的に向上する。
- 実用的なロボットを開発するには 学習・推論コンポーネント のように、データ(状況)から学習し、推論(判断)をおこなう機能が必ず必要となる。
- ハードウェアメーカーなどと協力し、より実用的な 学習・推論コンポーネント を開発していきたい。