

自然言語処理2014
-平成26年11月3日(No5)-

東京工科大学
コンピュータサイエンス学部
亀田弘之

前回の復習から

今日の課題

先週授業までであなたは何を学びましたか？
列挙しなさい。

解答例)

1. ○○を学んだ。
2. ○○のインストール手順
3. ○○の定義
4. ○○という用語

NLPのプログラムを書いてみよう！

I. Prologのインストール

II. プログラム作成手順

1. 解析対象テキストの収集
2. IC分析(直接構成素分析, Immediate Constituent Analysis)
3. 形式文法の設定
4. Prolog形式への書き換え
5. NLPプログラムの実行(その1)
6. 構文木を出力するプログラムへの拡張
7. NLPプログラムの実行(その2)

I. Prologのインストール

- (再確認)

Prologの種類

- Swi-prolog
- SICStus-Prolog
- I/F-Prolog
- K-Prolog
- Minerva
- Arity/Prolog
- RUN/Prolog
- C-Prolog
- Gnu Prolog
- Open Prolog
- YAP Prolog など

自由課題(提出の必要性なし)

- どんなProlog処理系があるのか調べてみよう。
- それぞれのWeb Siteを見つけてみよう。
- どれか1つdownloadして使ってみよう。
- 各Prologの特徴を比較し、一覧表に整理してみよう。

(注) 本授業ではswi-prologを使います。
なぜでしょうか？
理由を考えてみてください。



II. Prologによる構文解析プログラム

- 処理対象(1. 解析対象テキストの収集):

Tom broke the cup.

2. IC分析

- Tom broke the cup

文

————— —————
名詞句 動詞句

————— ————— —————
名詞 動詞 名詞句

—— ———
限定詞 名詞

3. 形式文法の設定

- 開始記号 = s
- 終端記号 = { tom, broke, the, cup }
- 非終端記号 = { s, n, v, d, np, vp }
- 書き換え規則 = {
 $s \rightarrow np + vp.$ $np \rightarrow n.$ $np \rightarrow d+n.$ $vp \rightarrow v+np.$
 $n \rightarrow tom.$ $n \rightarrow cup.$ $v \rightarrow broke.$ $d \rightarrow the.$
}

4. Prolog形式への書き換え

s :- np, vp.

np :- n.

np:-d, n.

vp:-v, np.

n:-tom.

n:-cup.

v:-broke.

d:-the.

Prog1.pl

s(A,C):-n(A,B),vp(B,C).

vp(A,C):-v(A,B),np(B,C).

np(A,C):-d(A,B),n(B,C).

n([tom | T],T).

n([cup | T],T).

v([broke | T],T).

d([the | T],T).

Prog2.pl

```
s (A, C, s (_n, _vp)) :- n (A, B, _n), vp (B, C, _vp).  
vp (A, C, vp (_v, _np)) :- v (A, B, _v), np (B, C, _np).  
np (A, C, np (_d, _n)) :- d (A, B, _d), n (B, C, _n).
```

```
n ([tom|T], T, n (tom)).  
n ([cup|T], T, n (cup)).  
v ([broke|T], T, v (broke)).  
d ([the|T], T, d (the)).
```

今日の内容

1. Prologにおけるユニフィケーションを理解する.
2. Prologの動作を理解する.
3. 構文解析プログラムの動作を理解する.
4. 構文解析プログラムを自分で書いてみる.
5. その結果、先週までの内容の理解を深める.

1. Prologにおける ユニフィケーションを理解する

- ① 定数と定数
- ② 定数と変数
- ③ 変数と変数
- ④ リスト

定数と定数

unifyする(同一視可)

- 2013 \Leftrightarrow 2013
- 3.14159 \Leftrightarrow 3.14159
- abcd \Leftrightarrow abcd

unifyしない(同一視不可)

- 2013 × \Leftrightarrow 1986
- 3 × \Leftrightarrow 3.0
- abcd × \Leftrightarrow xyz

定数と変数

unifyする(同一視可)

unifyしない(同一視不可)

変数と変数

unifyする(同一視可)

unifyしない(同一視不可)

リスト

unifyする(同一視可)

unifyしない(同一視不可)

実際の例

unifyする(同一視可)

unifyしない(同一視不可)

3. 構文解析プログラムの動作を理解

- (例文) *Tom broke the cup.*

4.構文解析プログラムを自分で書く

- 作成手順は？
(これも復習事項です。思い出せますか？)

作成手順の概要

1. 処理対象を決める
2. 処理対象の各文に対して、統語構造の分析を行う
3. 上記の分析結果をもとに、文法を書き下す
4. 上記で得られた文法を、Prologの形式に書きかえる
5. 動作を確認する

自習問題(提出の必要性なし)

- 例文)

Time flies like an arrow.

(アドバイス)教科書 “自然言語処理”, 天野 他, オーム社(2007).
第3章の3.2を参考にすると良いと思います。

3.その他

構文解析の種類

- 分類1
 - トップダウン解析方式(Topdown)
 - ボトムアップ解析方式(Bottom-up)
- 分類2
 - 縦型解析方式
 - 横型解析方式

確認問題

- 下記の用語を自分の言葉で説明しなさい。
 - 構文解析
 - トップダウン解析方式とボトムアップ解析方式
 - 縦型解析方式と横型解析方式
 - バックトラック
 - プログラミング言語Prolog

次回(9/10)の予告

- **日本語**を対象とするプログラムを一緒に作成します。
- PCを忘れないように。
- Swi-Prologが動くこと、**UTF-8コード**で保存できるエディターが必要です。メモ帳でもOKです。
- 必要なら、日本語の文法書や辞書を持参してください(たぶん要らないと思いますが...)。