HTML5 字句解析仕様の 自然言語処理による意味解析

17B01064 五十嵐 彩夏

東京工業大学 情報理工学院 数理・計算科学系

概要

あいうえお a

構文解析木

自然言語言語処理することによって, 得られる構文解析木は以下のようなものである.

HTML5 字句解析仕様

字句解析の各状態の記述の例

13.2.5.10 RCDATA end tag open state

Consume the next input character:

↪*ASCII alpha*

Create a new end tag token, set its tag name to the empty string. Reconsume in the RCDATA end tag name state.

↪*Anything else*

Emit a U+003C LESS-THAN SIGN character token and a U+002F SOLIDUS character token. Reconsume in the RCDATA state.

仕様記述言語

HTML5 字句解析仕様の記述内容をもとに, その仕様記述言語を以下の型として定義した.

- Command 命令文を表現する型
  - Bool 条件分岐文の条件部分を表現する型
- CommandValue 字句解析仕様の変数や値を表現する型
- InplementVariable 字句解析仕様の代入される変数を表現する型

それぞれの型が有する値は仕様書に出てくる文, 語句に基づいて定めた.
例えば, Command 型には以下のような値を持つ.

```
Command ::= If(b, cList1, cList2)   // if b then cList1 else cList2 (cLis
          | Switch(cval)            // cval が状態を表す値の時, 状態
          | Set(ival, cval)         // 変数 ival に値 cval を代入する (
          | Emit(cval)              // cval がトークンを表す値の時,
          | Create(string, cval)    // cval がトークンを表す値の時,
                                        トークン cval を新たに作り, 変
          ...
```

HTML5 字句解析仕様 2

字句解析の各状態の記述の例

13.2.5.10 RCDATA end tag open state

Consume the next input character:

↪*ASCII alpha*

Create a new end tag token, set its tag name to the empty string. Reconsume in the RCDATA end tag name state.

↪*Anything else*

Emit a U+003C LESS-THAN SIGN character token and a U+002F SOLIDUS character token. Reconsume in the RCDATA state.

ASCII alpha の部分の仕様記述言語のプログラム

Create(“1”, NewEndTagToken),
Set(INameOf(IVariable(“1”)), CString()),
Reconsume(StateName(RCDATA end tag name state))

自然言語処理

あいうえお a

Tag 型への変換

あいうえお a

文字列の前処理

字句解析仕様の文に対してそのまま自然言語処理を適用すると, トークンの分割や品詞解析が適切な形で解釈されない場合がある. そのため自然言語処理する際に前処理として, 特定の文字列の置き換えをし, 適切に文章が解釈されるようにする.
例えば, 以下のような置き換えを行う.

- 状態名を 1 つのトークンとして認識させるため, 字句解析の状態名を表す語句に対して, 空白 及び “-” を “_” に置き換える. “(” , “)” を除く. 先頭を大文字にする.
- ユニコード “U+xxxx” を 1 つのトークンとして認識させるため, “+” を “_” に置き換える.
- 命令文の解釈が上手くいくように, “Reconsume” など自然言語の解釈が上手くいかない動詞の前に仮の主語を表す “you” を加える.

Tag 型への変換の例

あいうえお a

仕様記述言語への変換

あいうえお a

実装

プログラミング言語 Scala で仕様記述言語のインタプリタを作成した. そして, 仕様記述言語のインタプリタと, 自然言語処理によって形式化した字句解析仕様をもとに, HTML5 の字句解析器を実装した.

字句解析器のテスト

HTML5 の字句解析のテストデータを使い, 字句解析器のテストを行った.

字句解析器のテスト

問題点書く解決方法書く

まとめ

HTML5 の字句解析仕様に対して, 自然言語処理の適用をし, その単語と構文木, 参照関係の解析結果を用いることで, 命令の自動形式化を行った.
正しく命令の抽出を行えたことを確認できた.
自然言語処理の適用において, 係り受け解析 (dependency parse) の情報の利用も検討したい.