概念辞書構築支援ツールの開発

上野正巳，唐澤 博
山梨大学

1 はじめに

階層構造を持った知識を作成するツールに与えるために，それを記述するための形式を規定し，それに従って知識を記述する必要がある。我々はフレーム理論の枠組みをPrologの上に構築したシステムCRSを用いて，これを実現している。概念辞書はまさにこの階層構造を持ち腐った知識であるが，従来この編集にはテキストエディタのみを用いて編集を行ってきた。また，CRSに導入されたsyntax sugarにより作業効率は良くなってきている。しかし，知識の量が多くなるほど，階層関係の直感的な把握がしづらい，他のユーティックの定義を参照のための操作が繁雑になるなど課題点は残っている。このような課題点を解決するために，概念辞書構築を支援するツールの作成を行なったので報告する。

2 ツールの構成

作業環境はX-Window System上に構築した。実際の実行画面を図1に示す。ツールは図2に示す構成になっており，構成要素は次のようになっている。

- 概念辞書定義ファイルを編集するエディタ
  エディタはXに対応し，プロセス間通信機能をサポートしているNEmacsを使用した。マクロを用いてブラウザ，コンバータとのデータのやり取りを行い，編集操作を支援する。

- 定義の階層構造を表示し，エディタへの操作ができるブラウザ
  ブラウザはXのクライアントとして作成した。受付するデータの形式に汎用性があるため，改造して他の目的にも使うことが可能である。

- 概念辞書定義ファイルからブラウザ用のデータへ変換を行うコンバータ
  コンバータはjgawkのスクリプトで記述した。

図1：実行例

Concept Dictionary Constructing Tool
Masami UENO, Hiroshi KARASAWA
Yamanashi Univ.
3 ツールの機能

ツールはエディタとブラウザ間で図3に示す命令をやりとりし、次のような機能を提供する。

- 概念辞書の階層関係を視覚的に確認しながら編集が行える
  エディタで編集中の概念辞書の階層構造をグラフィカルに表示する。階層構造の中には常に焦点となっているクラスが存在し、反転表示で示される。これを焦点ユニットと呼ぶことにする。

- 焦点ユニットからの表示の深さを対話的に調整できる
  階層構造の木が大きい場合に部分的に表示することで可読性をあげる。

- 編集するクラスをブラウザから選択できる
  マウスクリックでクラスを選択し、編集を指示するボタンをクリックすることにより、エディタのカーソルがクラスの定義位置へジャンプする。

- 焦点ユニットの定義内容を簡潔的に確認できる
  エディタに別ウィンドウを開き、選択したクラスの定義内容を表示する。

- 新たに選択した焦点ユニットを表示の中心にし、再描画できる
  再描画の指示はエディタに通知される。これを受けたエディタは、コンバータを用いて、指示された焦点ノード、表示の深さに従った階層関係のデータを新しくブラウザに送る。

ブラウザには上記機能のほかに編集内容の破棄、編集内容の保存を行うためのボタンを配置し、エディタに命令を発行することにより、対応する編集操作を行う。また、エディタからブラウザへ画像構造のデータを渡す際は次の手順で処理を行なっている。

- エディタからコンバータへ編集中の概念辞書、焦点ユニット、表示の深さを渡し、結果を受け取る。

ブラウザが、エディタ(NEmacs)に対して発行する命令

- (css-browser-ready)
- (css-browser-show-unit "クラス名")
- (css-browser-edit-class "クラス名")
- (css-browser-range レンジ)
- (css-browser-recenter)
- (css-browser-refocus "クラス名")
- (css-browser-quit-edit)
- (css-browser-save-kb)

エディタからブラウザに渡されるデータと命令

- cクラス 親クラス クラスの親子関係データ
- fクラス * 焦点ユニットの指示データ
- a* * データの終了
- eQuit ブラウザの終了
- eClear 階層表示の消去
- eScopeRange スコープレンジの通知

図3：エディタ・ブラウザ間の命令

ブラウザ側のデータ受信は割込みを用いて処理しているため、ブラウザとの同期を取りながらコンバータ後のデータをブラウザへ1行ずつ送る。

4 おわりに

ブラウザとエディタとの組み合わせで概念辞書の操作が実現できた。本ツールは辞書構造の比較的初期の段階をサポートするものであるが、今後、他の辞書とのリンクが増え構造が複雑になるに従い、文献[1]のようなハイパテキストの考えの導入が必要になってくると思われる。また、辞書をPrologシステムに読み込み、実際に利用しなければ取得できない情報を参照するために、編集中にPrologをバックグラウンドで走らせ、コンバータの動作の他に各クラスが継承している属性の表示などの高度な処理も行えるようにした。さらに、記述しやすい文法を導入してはいるが、相変わらずユーザは文法に配慮しながら作業を進める必要があるため、ユーザから概念辞書の文法を補完し、ユーザは概念辞書の構造のみに集中できるような編集環境を構築したい。

参考文献

[1] 安達、三輪、中沢、鈴木: 大規模電子化辞書開発における高機能辞書エディタ、第41回情処全大会05-5,p.3-195,(1990)