構文木付きコーパス作成支援統合環境: eBonsai

市川 宙† 野口 正樹‡ 吉田 恭祐‡
橋本 泰一‡ 徳永 健伸‡ 田中 穂積 ‡
† 東京工業大学 工学部 情報工学科
‡ 東京工業大学大学院 情報理工学研究科 計算工学専攻
{ichikawa,mnoguchi,rincho,taiichi,take,tanaka}@cl.cs.titech.ac.jp

1 概要

近年，自然言語処理の分野では，大規模な言語資源に基づく統計的手法が研究の中心となっている．言語資源の中でも，構文木付きコーパスは，特に統計的構文解析手法における重要な言語資源である．しかし，構文木付きコーパスは，人手により作成するため，多大な労力を必要とする．そのため，このようなコーパスの作成支援ツールの研究が行われるようになった [1, 2]．このようなツールの 1 つとして，構文木付きコーパス作成支援統合環境 **eBonsai**（図 **1**）を開発した．

eBonsai は，統合環境 Eclipse 上に，以下の 2 つのプラグインを実装したものである．

- アノテーションプラグイン．このプラグインは，ユーザが正しい構文木の候補から，正しい構文木を視覚的に選択する作業を支援する．
- 構文構造検索プラグイン．このプラグインは，構文木をクエリとし，クエリと同じ構造を含む文をコーパスから検索するために利用する．このプラグインにより，ユーザは，構文構造を付与する際に，参考となる文をコーパスから容易に検索できるようになる．

この 2 つのプラグインは，Eclipse 上で連携して動作する（例えば，アノテーションプラグインから構文構造検索プラグインに木構造をコピー&ペーストできる）．また，Eclipse が持つプロジェクト管理機能や，その他のプラグイン（CVS 機能など）と組み合わせる事で，Eclipse をコーパス作成の統合環境として利用できるようになった．

2 構文木付きコーパスの作成方法

eBonsai を使った構文木付きコーパス作成の流れは，以下のとおりである（図 **2**）．

1. 平文コーパスから文を取り出す．
2. その文を MSLR パーザ [3] で構文解析する．

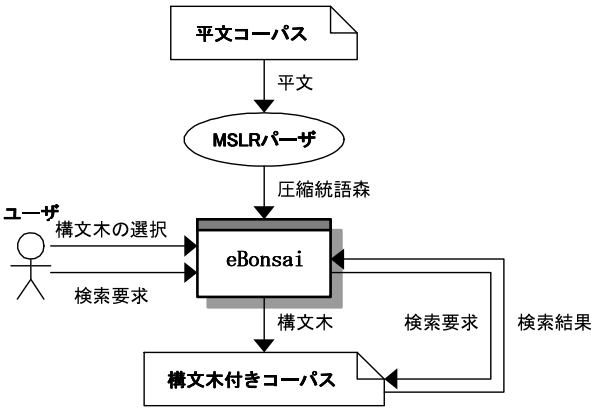


図 2: eBonsai を使った構文木付きコーパス作成の流れ

3. 得られた構文木の集合の中から，アノテーションプラグインを使って，正しい構文木を選ぶ．
   この時，構文構造検索プラグインを使って，他の文の構文木を参考にする事ができる．
4. 選んだ構文木を構文木付きコーパスに加える．

MSLR パーザでは，多様な日本語の文をカバーするため，適用範囲の広い文法が用いられる．そのため，2 の段階で構文木が一意に決まる事は滅多になく，2 では構文木の候補の集合が（圧縮統語森の形式で）出力される．その候補の中から正しい構文木を選ぶのが 3 の作業であり，ここで eBonsai が使用される．

3 アノテーションプラグイン

3.1 アノテーションプラグインの概要

アノテーションプラグインは，MSLR パーザの出力である圧縮統語森（構文木の候補の集合）から正しい構文木を選択するためのツールである．

構文木の候補は，場合によっては数万にもなるため，それらを 1 つ 1 つ見て正しいものを探すのは時間を要する．そこで，候補の 1 つを視覚的に表示し，正しい構文木が満たすべき制約をユーザが繰り返し指定することで，候補を絞り込み，正しい構文木を容易に選択

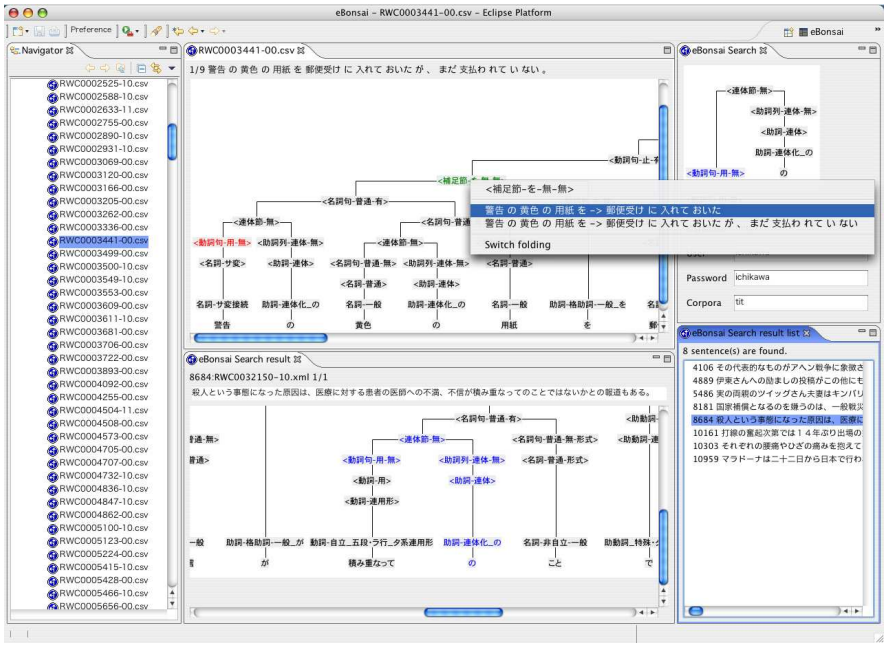


図 1: eBonsai

できるようになっている.「満たすべき制約の指定」には，以下の 2 種類がある.

1. 係り受けの指定．どの節（連体節，連用節など）がどの句（名詞句，動詞句など）に係るのかを指定する.
2. 非終端記号の指定．ノードが動詞句なのか名詞句なのか，などを指定する.

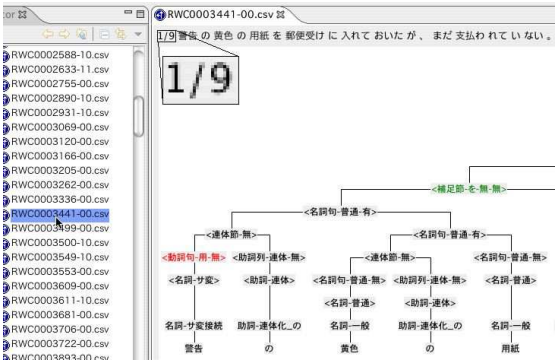
このプラグインは，岡崎 [4] のツールを Eclipse プラグインとして再実装したものである.

3.2 アノテーションプラグインの操作方法

以下の文を例にとって，アノテーションプラグインの使い方を説明する.

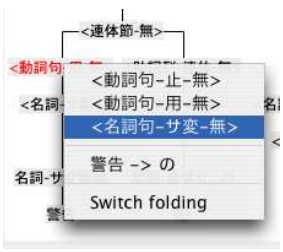
> 警告の黄色の用紙を郵便受けに入れておいたが，まだ支払われていない.

1. Eclipse の「Navigator」上で，この文の圧縮統語森ファイル（RWC0003441-00.csv）をダブルクリックする.
2. 新しくウィンドウが開き，木構造の候補の 1 つが視覚的に表示される.

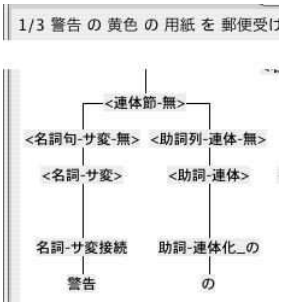


このウィンドウをアノテーションエディタと呼ぶ. 左上には候補の総数 9 が表示されている.

3. ”< 動詞句-用-無 >”のノードが赤くなっている. これは,「非終端記号の候補が複数ある」という意味である.
4. ”< 動詞句-用-無 >”を右クリックすると，ポップアップメニューに非終端記号の候補一覧が表示される.

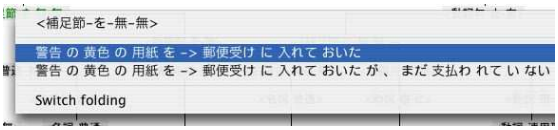


5. ここの「警告」は明らかに名詞句なので，”< 名詞句-サ変-無 >”を選ぶ．
6. すると，”< 動詞句-用-無 >”が”< 名詞句-サ変-無 >”に変わって黒くなり，このノードの曖昧性が解消された事が分かる．

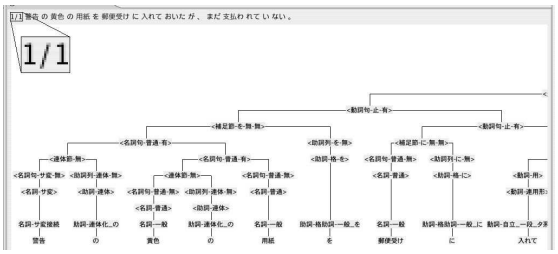


左上の候補総数も 3 に減少している．

7. 次に，”< 補足節-を-無-無 >”のノードが緑になっている．これは,「係り先の候補が複数ある」という意味である．
8. ”< 補足節-を-無-無 >”を右クリックすると，ポップアップメニューに係り先の候補一覧が表示される．



9. 「警告の黄色の用紙を まだ支払われていない」というのは変なので，ここでは上の候補を選ぶ．
10. ノードが全て黒くなり，候補総数が 1 になった．



これでこの文のアノテーションは終了である．

3.3 アノテーションプラグインのその他の機能

他にも以下の機能が実装されている．

図 3: アノテーションエディタの右クリックメニュー

- 回数無制限の Undo/Redo．結果を保存して終了したあとも，Undo/Redo が可能．アノテーションエディタの右クリックメニュー（図 **3**）から行う．
- 他の木構造候補の表示．アノテーションエディタの右クリックメニューの [Previous tree] [Next tree] で行う．
- 枝の畳み込み．ノードを右クリックして [Switch folding] を選ぶと，そのノード以下の枝を畳み込んで表示する．
- 木構造の一部を，構文構造検索プラグインにコピーする．アノテーションエディタの右クリックメニューの [Copy to search] で行う．後述．

4 構文構造検索プラグイン

4.1 構文構造検索プラグインの概要

アノテーションプラグインを使った作業で，ユーザは係り先や非終端記号の選択を求められる．多くの場合，文意を考えれば，これらの選択は容易である．しかし，選択に迷うような微妙なケースも存在する．そのような時にヒントを提供するのが，構文構造検索プラグインである．

構文構造検索プラグインは，クエリとして木構造を与え，その木構造を含む文をコーパスから検索し，表示する．

構文構造の検索には，吉田 [5] の手法を用いている．

4.2 構文構造検索プラグインの操作

1. クエリ入力ウィンドウ（図 **4** 左上）でクエリとなる木構造を作る．
   - ノードのラベルを変更するには，ノードを左クリックして選択し，下の入力欄に入力する．ラベル名にはワイルドカード % も使える．
   - 子ノードを追加するには，ノードを右クリックして [子の追加] を選ぶ．

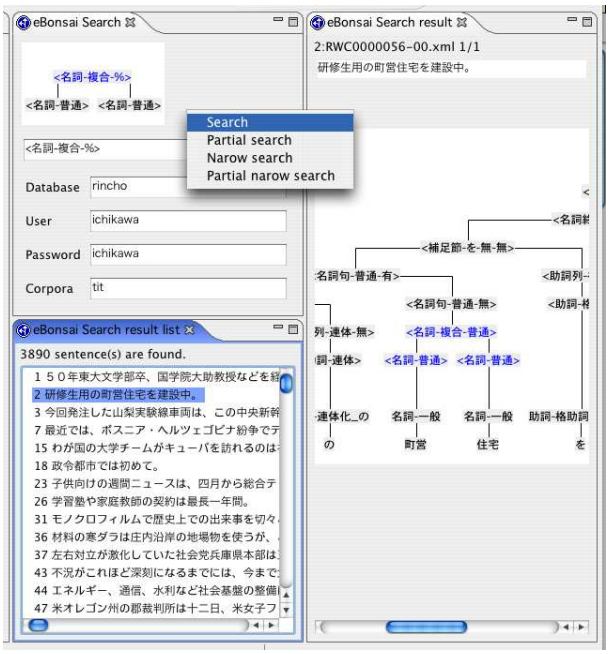


図 4: 構文構造検索プラグイン

2. クエリ入力ウィンドウを右クリックし，[Search] を選ぶ．
   - [Partial search] を選ぶと，子の数がクエリと一致しなくてもヒットするようになる．
   - [Narrow search] を選ぶと，前回の検索結果から絞り込む事ができる．
   - [Partial narrow search] は Partial search と Narrow search の組み合わせである．
3. 検索結果一覧ウィンドウ（図 4 左下）にヒットした文が一覧表示される．
4. 一覧中の文をクリックすると，検索結果詳細ウィンドウ（図 4 右）に選んだ文の木構造が表示される．ヒットした部分は中央に青く表示される．
5. 同じ文の複数箇所にヒットした場合は，検索結果詳細ウィンドウに 1/2 などと表示されるので，右クリックメニューで [Previous match] [Next match] を選んで，他のヒット箇所を表示できる．

5 アノテーションプラグインと構文構造検索プラグインの連携

以下の手順で，アノテーションプラグインで表示されている木構造の一部を，構文構造検索プラグインのクエリ入力ウィンドウにコピーできる．

1. アノテーションプラグイン上で左ドラッグし，木構造の一部を選択する（ノードが青くなる）．
2. アノテーションプラグイン上で右クリックし，[Copy to search] を選ぶ．クエリ入力ウィンドウに，選択中の木構造がコピーされる．
3. 必要なら，クエリを修正する．
4. クエリ入力ウィンドウを右クリックし，[Search] などを選んで検索する．

先ほどの例なら，この機能を使って，どういう場合に「動詞句+助詞”の”」になるのかを調べる事ができる．

6 まとめと今後の課題

本論文では，構文木付きコーパスの作成を支援する統合環境 eBonsai を紹介した．eBonsai によって，構文木付きコーパスの作成作業を，統合環境の中でシームレスに行えるようになった．

今後の課題として，以下のようなものが挙げられる．

- 本格的な構文付きコーパス作成作業に適用．
- CVS などを利用した共同作業の方針を確立．
- インタフェースを改善．
- ヒントとなる情報（参考になる文の木構造，特例的な作業方針など）を自動表示．

参考文献

[1] Oliver Plaehn, Thorsten Brants. Annotate – An Efficient Interactive Annotation Tool. Proceedings of the Sixth Conference on Applied Natural Language Processing ANLP-2000, Seattle, WA, 2000

[2] Proceedings of the Third International Conference on Language Resources and Evaluation, Paris: European Language Resources Association, 2002

[3] 白井清昭, 植木正裕, 橋本泰一, 徳永健伸, 田中 穂積. 自然言語解析のための MSLR パーザ・ツールキット. 自然言語処理. Vol.7. No.5. pp.93 – 112. 2000. Nov

[4] 岡崎篤, 白井清昭, 徳永健伸, 田中穂積. 正しい構文木の選択を支援する構文木付きコーパス作成ツール. 人工知能学会 第 15 回全国大会, 2001

[5] Kyosuke Yoshida, Taiichi Hashimoto, Takenobu Tokunaga, Hozumi Tanaka. Retrieving Annotated Corpora for Corpus Annotation. Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004), 2004