単語単位の日本語係り受け解析

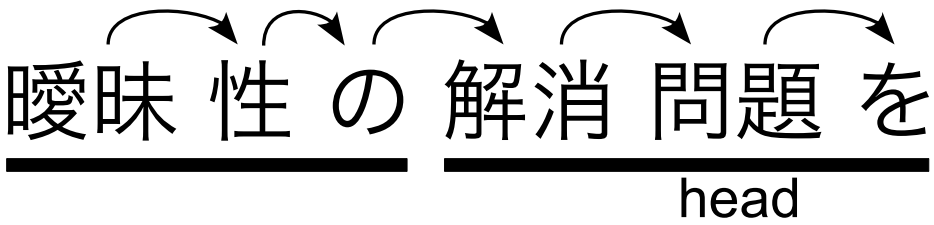
Daniel FLANNERY[1] 宮尾 祐介[2]
Graham NEUBIG[1] 森 信介[1]

[1] 京都大学 情報学研究科　　[2] 国立情報学研究所

2012 年 3 月 16 日

日本語の係り受け解析

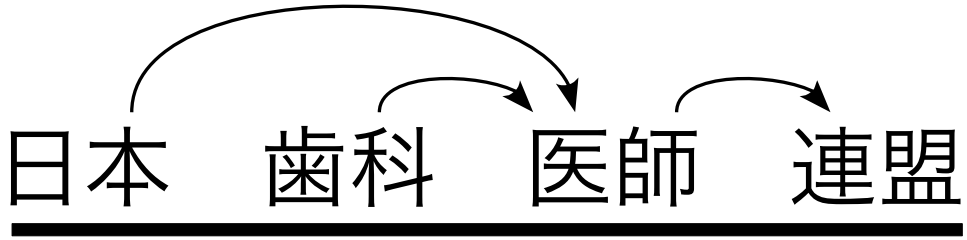
- 文節を単位とするのが主流
- しかし、文節は日本語特有の単位
- より細かく分析しなければならない場合もある



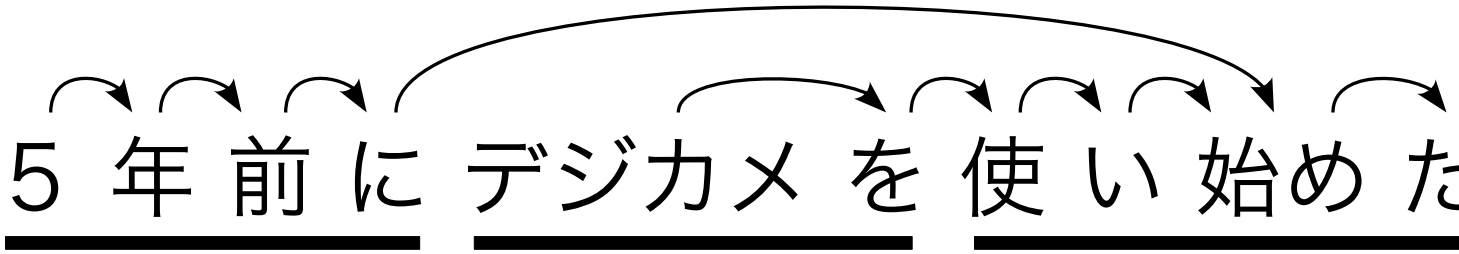
  - 複合語の構造
- 提案手法のポイント
  - 部分的アノテーションによる迅速分野適応
  - 様々な言語資源が利用可能
  - 交差する係り受けも扱える

単語を単位とする利点

- **複合語内の構造がわかる**



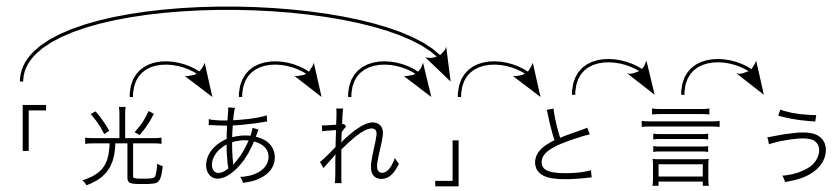
  - **省略語の候補生成 (「日医連」ではなくて「日歯連」)**



  - **「5 年前に始めた」と「デジカメを使う」という修飾関係が分かる**

その他の利点

- 括弧や引用符などの記号対の対応が示せる

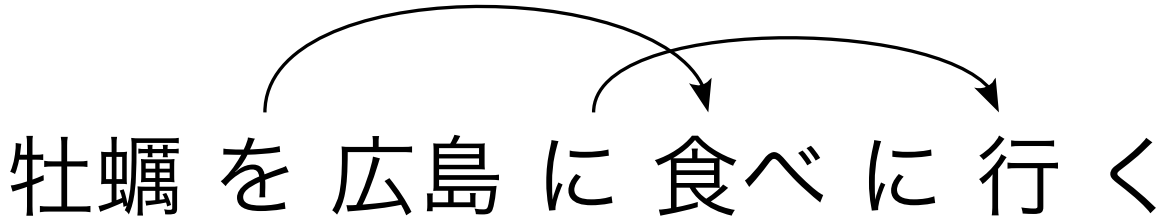


  - 自明な対応から教師なし学習が可能

- 他の言語との親和性
  - 他の言語では単語単位が一般的
  - 機械翻訳

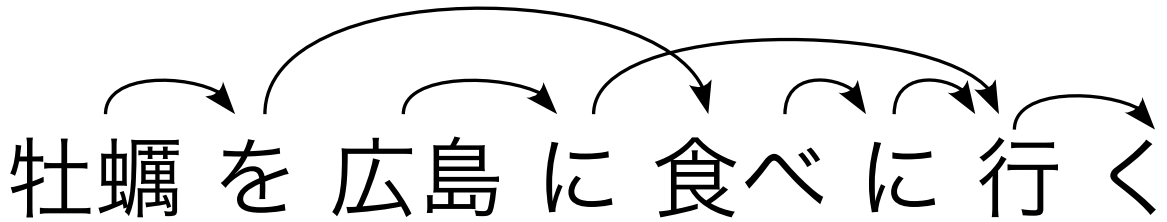
部分的アノテーションによる迅速分野適応

- **部分的アノテーションコーパスが利用可能**



  - **分野特有の表現のみ情報付与**
  - **点予測：周辺の係り受け情報を参照しない**

- Cf. **フルアノテーションコーパス**
  - **全ての単語の係り先を付与**
  - **周辺の係り受け情報が扱いやすい**

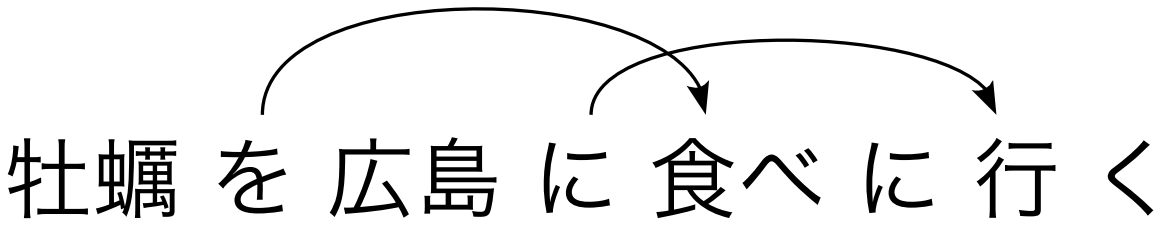


問題点

- 文節係り受けよりアノテーションが多くなる
- 係り先の付与が難しい場合がある
- 交差する係り受けがより起こりやすい

問題点

- 文節係り受けよりアノテーションが多くなる
  - 部分的アノテーションで回避
- 係り先の付与が難しい場合がある
  - 部分的アノテーションで回避
- 交差する係り受けがより起こりやすい
  - 提案手法はこのような係り受けも扱える



係り受け解析

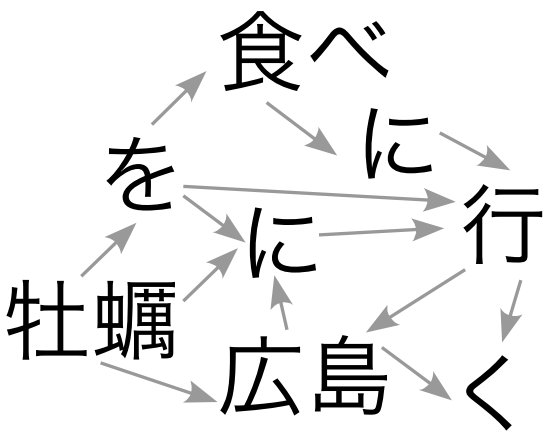
1. 単語をノードとする
2. 修飾関係を枝とする
3. 木構造

- 入力：単語列 $\vec{\mathbf{w}} = \langle w_1, w_2, \ldots, w_n \rangle$
- 出力：$\vec{\mathbf{d}} = \langle d_1, d_2, \ldots, d_n \rangle$、$d_i$ は w_i の係り先
- 日本語の書き言葉：係り先は必ず右にある

最大全域木による係り受け解析 [McDonald et al., 2005]

1. **ある定義でエッジスコア $\sigma(\langle i, d_i \rangle, \vec{\mathbf{w}})$ を計算**
2. **以下の最大全域木を探索**

$$\hat{\vec{\mathbf{d}}} = \underset{\vec{\mathbf{d}} \in D}{\mathbf{argmax}} \sum_{i=1}^{n} \sigma(\langle i, d_i \rangle, \vec{\mathbf{w}})$$

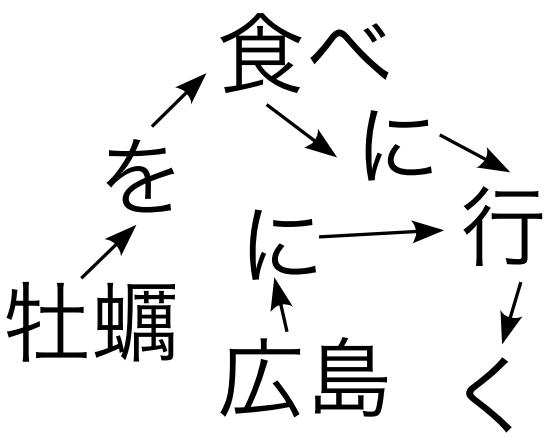


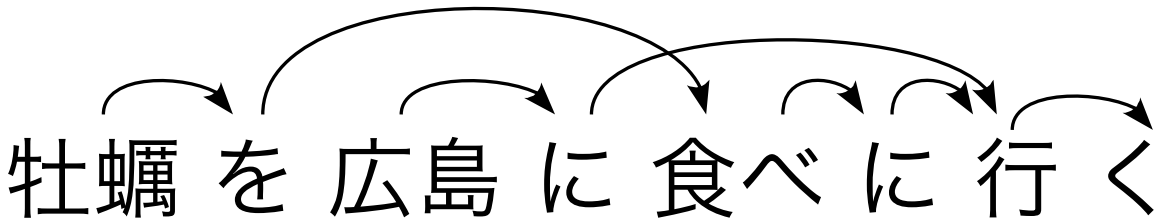
牡蠣 を 広島 に 食べ に 行 く

最大全域木による係り受け解析 [McDonald et al., 2005]

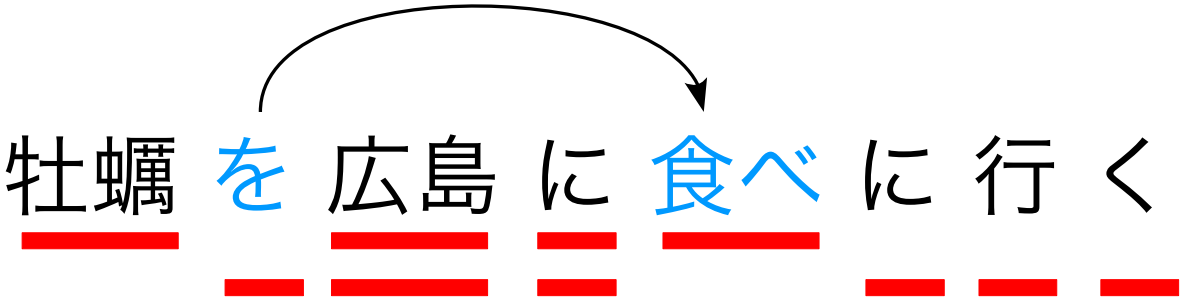
1. **ある定義でエッジスコア $\sigma(\langle i, d_i \rangle, \vec{\mathbf{w}})$ を計算**
2. **以下の最大全域木を探索**

$$\hat{\vec{\mathbf{d}}} = \underset{\vec{\mathbf{d}} \in D}{\mathbf{argmax}} \sum_{i=1}^{n} \sigma(\langle i, d_i \rangle, \vec{\mathbf{w}})$$





点予測によるエッジスコア計算 [IJCNLP 2011]

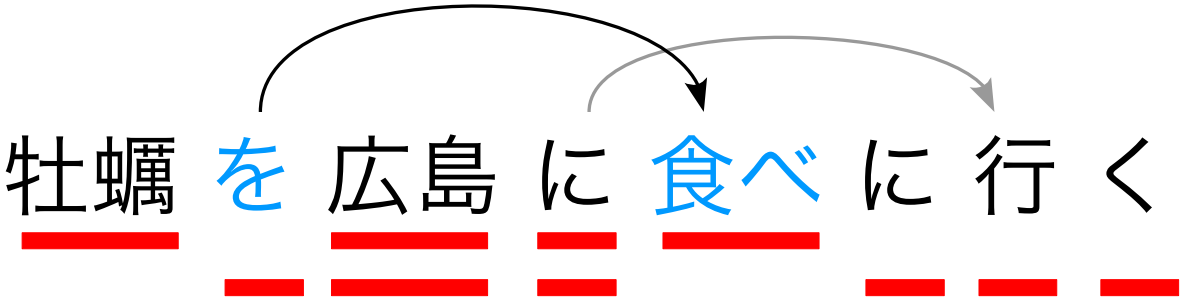


- 係り先の選択は多クラス分類問題

$$\sigma(\langle i, d_i \rangle) = p(d_i | \vec{\mathbf{w}}, i), \quad (d_i \in [0, n] \wedge d_i \neq i)$$

- それぞれの単語のエッジスコアを個別に計算する
- 素性
  1. 係り元と係り先の距離
  2. 係り元/係り先の表記
  3. 係り元/係り先の前後 3 単語

点予測によるエッジスコア計算 [IJCNLP 2011]



- 係り先の選択は多クラス分類問題

$$\sigma(\langle i, d_i \rangle) = p(d_i | \vec{\mathbf{w}}, i), \quad (d_i \in [0, n] \wedge d_i \neq i)$$

- それぞれの単語のエッジスコアを個別に計算する
- 素性
  1. 係り元と係り先の距離
  2. 係り元/係り先の表記
  3. 係り元/係り先の前後 3 単語
  4. ~~周辺の係り受け情報~~

評価

- 係り受け解析の精度を比較

1. 既存手法との比較
   - フルアノテーションコーパスでの学習
   - 同一分野のテスト
2. 分野適応
   - 一般分野のフルアノテーションコーパスと適応分野の部分的アノテーションコーパスでの学習
   - 適応分野のテスト
3. 文節を単位とする係り受け解析器との比較
   - フルアノテーションコーパスでの学習
   - 同一分野のテスト
   - 単語係り受けを文節係り受けに変換

コーパスの諸元

ID	種類	用途	文数	単語数	平均文長
EHJ-train EHJ-test	辞書例文	学習 評価	11,700 1,300	145,925 16,348	12.48 単語
NKN-test	新聞記事	評価	1,002	29,038	28.98 単語
EDR-train	新聞記事や雑誌など	PA プール	39,832	846,664	21.26 単語

- 単語は BCCWJ の短単位 (+活用語尾の分割)
- 品詞なし（BCCWJ から学習した KyTea [Neubig et al., 2011] で自動付与）
- EDR-train は部分的アノテーションのプール

1. 一般分野コーパスの利用 [NL201]

1. Malt: MaltParser [Nivre et al., 2006]
2. MST: MST Parser [McDonald et al., 2005]
3. EDA: **提案手法**

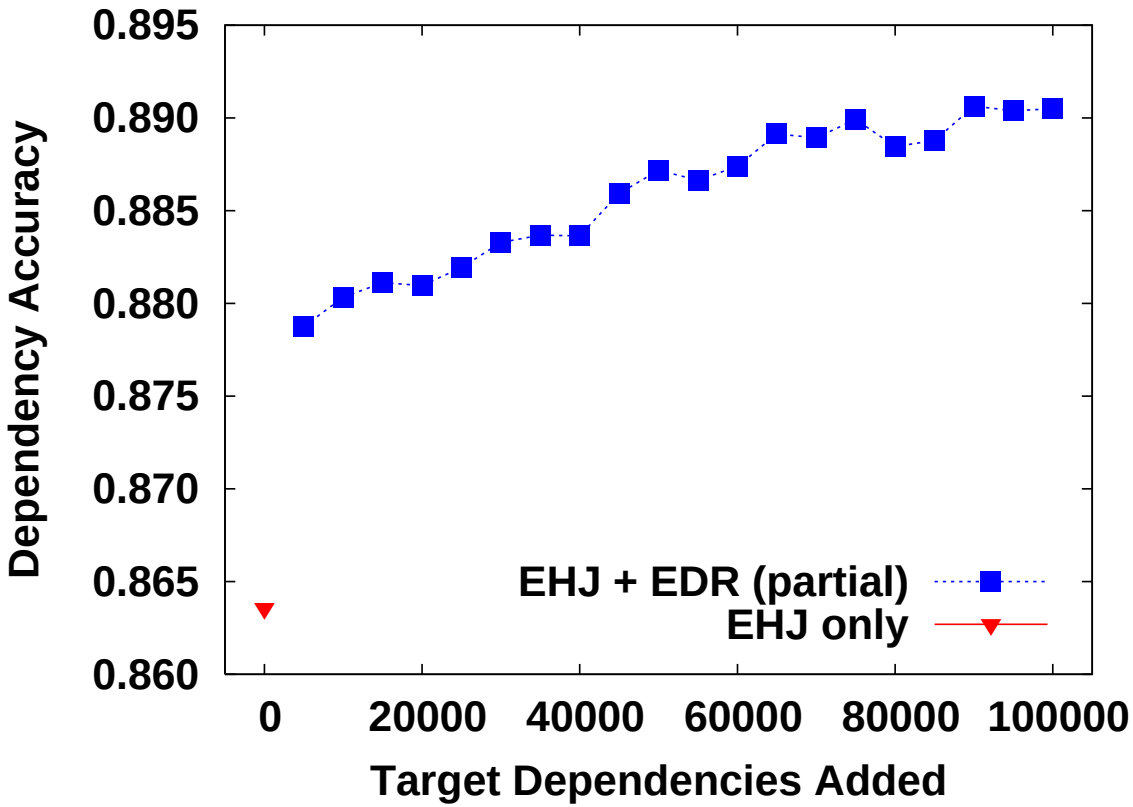
手法	係り受け精度
Malt	96.63%
MST	96.67%
EDA	96.83%

2. 部分的アノテーションによる分野適応

- 一般分野のフルアノテーションコーパス
- 適応分野の部分的アノテーションコーパス
- 目的
  - 部分的アノテーションコーパスからの学習
  - フルアノテーションコーパスとの比較
  - 適応分野に近いコーパスの利用

2. 部分的アノテーションによる分野適応

- 適応分野 (新聞記事) で評価
  1. EHJ の全部と EDR からの部分的アノテーション
  2. EHJ のみ



- 部分的アノテーションコーパスからの学習が可能

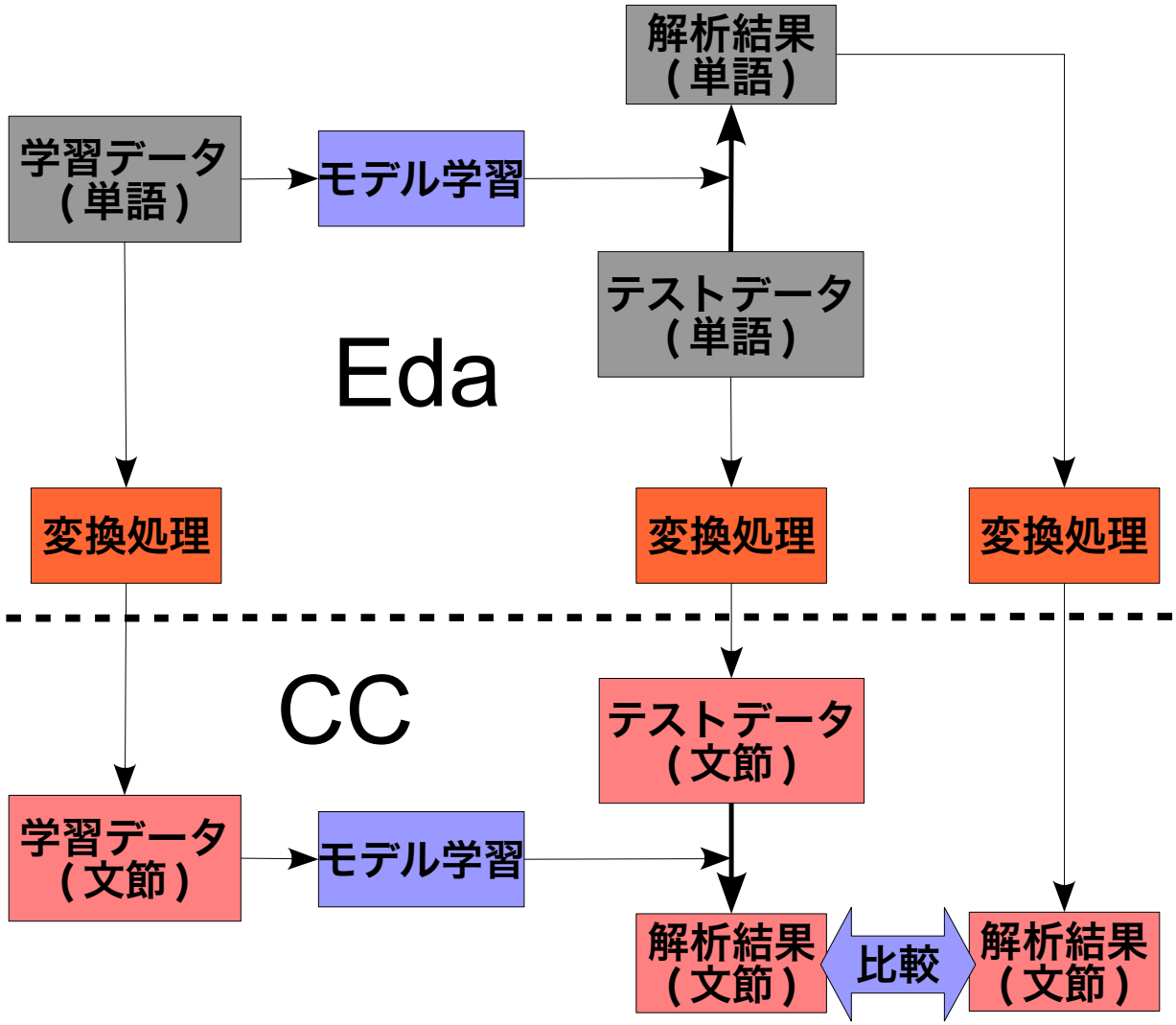
3. 文節係り受け解析器との比較

- **単語係り受けを文節係り受けに変換するルール**
  1. **単語を文節にまとめる**
  2. **単語係り受けから文節係り受けを同定**
- **一般分野のフルアノテーションコーパスを文節に変換**

ID	種類	用途	分数	単語数	文節数
EHJ-train	辞書例文	学習	11,700	145,925	53,694
EHJ-test		評価	1,300	16,348	6,024

- **比較手法**
  1. CC:**チャンキングの段階適用** [Kudo and Matsumoto, 2002]
  2. EDA:**提案手法**

3. 係り受け変換処理の流れ



3. 文節を単位とする係り受け解析器との比較

手法	文節係り受け精度	文正解率
CC	92.4%	72.5%
EDA	94.5%	78.2%

- 文節係り受け精度:全ての文節のうち、係り受け解析結果が正解と一致する割合
- 文正解率:係り受け解析結果が正解と一致する文の割合
- 注意
  - 単語係り受けだと学習事例が多い
  - 品詞大分類しか使っていない

まとめ

- 単語を単位とする係り受け解析
  1. フルアノテーションの利用では従来手法と同等の精度
  2. 部分的アノテーションによる分野適応
  3. 文節係り受け解析器と同等の文節精度
- EDA をオープンソースとして公開

http://www.ar.media.kyoto-u.ac.jp/members/flannery/eda/

単語係り受けコーパスの作成状況

- フルアノテーション (主にテスト用)

	コーパス	文数	状況
BCCWJ	Yahoo!知恵袋 (OC)	250	予定
	白書 (OW)	250	途中
	Yahoo!ブログ (OY)	250	予定
	書籍 (PB)	250	途中
	雑誌 (PM)	250	途中
	新聞 (PN)	250	途中
その他	NTCIR 特許	500	完成
	論文抄録	355	途中
	レシピ	1230	途中
	Wikipedia 京都関連	1235	完成

- それぞれの分野の部分的アノテーションコーパスも作成予定

参考文献

Flannery, D., Miayo, Y., Neubig, G., and Mori, S. (2011). Training dependency parsers from partially annotated corpora. In Proceedings of the 5th International Joint Conference on Natural Language Processing, pages 776–784.

Kudo, T. and Matsumoto, Y. (2002). Japanese dependency analysis using cascaded chunking. In Proceedings of the Sixth Conference on Computational Natural Language Learning, volume 25, pages 1–7.

McDonald, R., Pereira, F., Ribarov, K., and Hajič, J. (2005). Non-projective dependency parsing using spanning tree algorithms. In Proceedings of the 2005 Conference on Empirical Methods in Natural Language Processing, pages 523–530.

Neubig, G., Nakata, Y., and Mori, S. (2011). Pointwise prediction for robust, adaptable Japanese morphological analysis. In The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies Short Paper Track.

Nivre, J., Hall, J., and Nilsson, J. (2006). Maltparser: A data-driven parser-generator for dependency parsing. In Proceedings of the Fifth International Conference on Language Resources and Evaluation.