2V-5

カテゴリ階層を考慮した固有表現抽出

東山 翔平 † Mathieu Blondel †† 関 和広 †† 上原 邦昭 ††
† 神戸大学工学部情報知能工学科 †† 神戸大学大学院システム情報学研究科

1 はじめに

固有表現抽出は，テキスト中に現れる人名などの語句の同定を目的とする自然言語処理の基本的な問題である．抽出する固有表現は，人名や組織名など 10 種類程度を対象とすることが一般的であり，これらの固有表現カテゴリの間の関係は考慮しないことが多い [1]．しかし，これらのカテゴリは階層性を有する場合がある．たとえば，組織名はさらに会社名や大学名などのカテゴリに細分化されうる．このようなカテゴリの階層構造を考えたとき，階層的に近い（遠い）という情報は固有表現抽出の際に活用できる可能性があり，カテゴリ数が多い場合に特に有用であると考えられる．

本研究では，カテゴリの階層性を考慮した固有表現抽出法を提案し，その有効性を検証する．

2 関連研究

機械学習において，出力が構造を有する問題を構造学習という．最も単純な構造学習である系列ラベリングを行う手法として，隠れマルコフモデル（HMM）や条件付き確率場（CRF）がよく利用されてきた．Collin [2] が初めて，それらに比べてアルゴリズムが簡単であるパーセプトロンを系列ラベリング問題に適用し，最大エントロピー法（ME）を上回る結果を示した．また，Dekel [3] らは，出力が木構造を有する場合を対象とするオンライン学習およびバッチ学習の手法を提案した．

3 提案手法

固有表現抽出タスクは，入力文，すなわち単語の列に対して，固有表現タグの列を出力する系列ラベリングの問題と見なせる．本研究では，固有表現部分を同定する際のチャンクタグとしては IOB2 タグを用い，タグ列の推定には構造化パーセプトロンを用いる．

3.1 構造化パーセプトロンによる固有表現抽出

構造化パーセプトロンでは，単語列 $\boldsymbol{x}$ に対するタグ列 $\boldsymbol{y}$ のスコアをパーセプトロンの重みベクトル $\boldsymbol{w}$ と素性ベクトルとの内積 $\langle \boldsymbol{w}, \boldsymbol{\Phi}(\boldsymbol{x}, \boldsymbol{y}) \rangle$ で与える．入力 $\boldsymbol{x}$ を得ると，式 (1) を満たすタグ列 $\hat{\boldsymbol{y}}$ を予測ラベルとして出力する．

$$\hat{\boldsymbol{y}} = \underset{\boldsymbol{y} \in \mathcal{Y}^m}{\operatorname{argmax}} \langle \boldsymbol{w}, \boldsymbol{\Phi}(\boldsymbol{x}, \boldsymbol{y}) \rangle \qquad (1)$$

ここで，$\boldsymbol{\Phi}$ は単語列 $\boldsymbol{x}$ およびタグ列 $\boldsymbol{y}$ に関する素性関数，$\mathcal{Y} = \mathcal{Y}_0 \cup \{\mathrm{O}\}$ は固有表現および非固有表現タグの集合，m は単語列 $\boldsymbol{x}$ の長さである．ただし，$\mathcal{Y}_0 = \{e_1, \ldots, e_k\}$ は固有表現タグの集合，O は固有表現でないことを表すタグ（非固有表現タグ）である．

式 (1) は，k^m 個のラベル $\boldsymbol{y} \in \mathcal{Y}^m$ の中から，内積を最大にするものを求める k^m クラスの多値分類問題と考えることができる．ただし，m の値が大きくなるとクラス数が膨大になりやすいという計算量の問題がある．この問題を解消するため，動的計画法の一種であるビタビアルゴリズムを用いる．タグ列は単純マルコフ連鎖で生成されることを仮定し，これに応じて素性関数と損失関数を分割する．

素性は，単語列 $\boldsymbol{x}$ およびタグ列 $\boldsymbol{y}$ 中の同じ位置 i における単語とタグの組 $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ と，タグ列中の連続する位置 $i-1, i$ にあるタグの組 $(\boldsymbol{y}_{i-1}, \boldsymbol{y}_i)$ を用いる．

3.2 損失関数の導入

2 つのタグ $t, t' \in \mathcal{Y}$ について，階層構造において，これらが互いに近い位置にある場合に小さい値をとる損失関数 ℓ を考える．ℓ のとる値は非負の実数値である．階層構造は木構造であるとし，これらのタグはその葉ノードに相当するものとする．次に，2 つのタグ列 $\boldsymbol{y}, \boldsymbol{y}' \in \mathcal{Y}^m$ の間の損失関数 L を式 (2) で定義する．

$$L(\boldsymbol{y}, \boldsymbol{y}') = \sum_i \ell(\boldsymbol{y}_i, \boldsymbol{y}'_i) \qquad (2)$$

訓練データ中の事例 $\boldsymbol{x}$ に対するラベルの推定においては，推定された予測ラベル $\hat{\boldsymbol{y}}$ が正解ラベル $\boldsymbol{y}^\star$ と異なる場合に重み $\boldsymbol{w}$ を更新することで，重みの学習を行う．この際，式 (2) の損失関数を用いて，式 (3) によりタグの階層性を考慮したラベルの推定を行う．

$$\hat{\boldsymbol{y}} = \underset{\boldsymbol{y} \in \mathcal{Y}^m}{\operatorname{argmax}} \{ \langle \boldsymbol{w}, \boldsymbol{\Phi}(\boldsymbol{x}, \boldsymbol{y}) \rangle - \alpha L(\boldsymbol{y}^\star, \boldsymbol{y}) \} \qquad (3)$$

ここで，α は損失関数に対するハイパーパラメータであり，非負の実数値をとる．式 (3) では，0 または負の値をとる $-\alpha L(\boldsymbol{y}^\star, \boldsymbol{y})$ という項が追加され，ラベル $\boldsymbol{y}$ が正解ラベルから遠いほど，そのスコアは式 (1) の場合に比べて小さくなる．

3.3 損失関数の定義

タグについての損失関数 ℓ を具体的に定義する必要がある．本研究では，その損失関数を提案する．まず，

Named Entity Recognition Using Category Hierarchy
Shohei Higashiyama, Department of Computer Science and Systems Engineering, Kobe University (†)
Kazuhiro Seki, Kuniaki Uehara, Graduate School of System Informatics, Kobe University (††)

固有表現タグ $e, e' \in \mathcal{Y}_0$ の間の損失関数 ℓ_0 を式 (4) で定義する．

$$\ell_0(e, e') = \begin{cases} 0 & (e = e') \\ \frac{\max\{d(e,dca), d(e',dca)\}}{depth(dca)} & (e \neq e') \end{cases} \quad (4)$$

ここで，dca はタグ e, e' の共通の祖先のうち深さが最大であるノードを表す．d は 2 つのノードのグラフ上の距離を与える関数であり，$depth$ はノードの深さ，すなわち根ノードとの距離を与える関数である．e, e' と dca との距離が小さく，dca の深さが大きいほど ℓ_0 は小さい値をとる．

次に，式 (4) を用いて，非固有表現を含めたタグ $t, t' \in \mathcal{Y}$ に対する損失関数 ℓ を式 (5) で定義する．右辺の一番下の式は階層構造に依存する定数である．t と t' の一方のみが O である場合がこれに相当し，このときに ℓ は最大値をとる．

$$\ell(t, t') = \begin{cases} \ell_0(t, t') & (t, t' \in \mathcal{Y}_0) \\ 0 & (t = t' = \mathrm{O}) \\ \max\limits_{e,e' \in \mathcal{Y}_0} \{\ell_0(e, e')\} + 1 & (otherwise) \end{cases} \quad (5)$$

階層構造の例を図 1 に示す．左図は固有表現タグの元の階層構造である．この階層に，O タグとノード is_entity を追加したものが右図である．このように階層構造を拡張することで，式 (4) における e, e' の dca について，$depth(dca) > 0$ が満たされる．

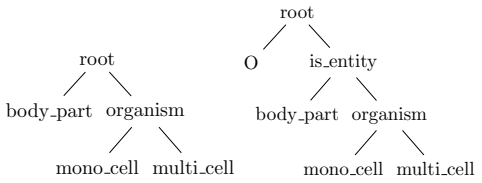

図 1: 固有表現タグの階層構造（左）とその拡張（右）

4 評価実験

4.1 実験データ

実験には，GENIA コーパスを利用した†．GENIA コーパスは，分子生物学の論文アブストラクトに専門用語情報等をタグ付けしたコーパスであり，固有表現タグの種類は 36，階層の最大の深さは 7 である††．GENIA コーパスでは，1 つの単語に複数の固有表現タグが付与されていることがある．たとえば，<<IL-2 gene> expression> では，外側の括弧内の単語にタグ other_name が付与され，内側の括弧内の単語にタグ DNA_domain_or_region が付与されている．本研究では，最も外側にあるタグのみを抽出の対象とした．

† http://www-tsujii.is.s.u-tokyo.ac.jp/~genia/corpus/GENIAcorpus3.02.tgz から入手可能．

†† 現在入手可能なコーパスと，GENIA Project のウェブサイトで公開されている固有表現タグの階層構造図とのバージョンが異なっており，実験には筆者らが定義した固有表現タグの階層構造を用いた．

4.2 実験設定

訓練データの学習において，式 (3) におけるハイパーパラメータ α の値を変化させることにより損失関数の影響を調べた．$\alpha = 0$ が損失関数を用いない場合（式 (1)）に相当する．正解ラベルが未知であるテストデータに対するラベルの推定においては，$\alpha = 0, \alpha > 0$ の場合とも式 (1) を用いた．

評価は，それぞれの α の値において以下のように行った．GENIA コーパスで訓練 4，テスト 1 の比率で 5 分割交差検定を行い，5 回の精度，F 値（$\beta = 1$）および損失値の平均を算出した．損失値とは，テスト事例に対する損失関数（式 (2)）の値を全テストデータについて足し合わせ，出現単語数で割って正規化した値である．損失値は値が小さいほど良い．実験に使用した GENIA コーパスの階層構造では，この値の範囲は最小で 0，最大で 7 である．訓練データの学習は 5 回行った．

4.3 実験結果と考察

実験結果を表 1 に示す．階層性を考慮しない評価尺度である精度，F 値と，階層性を考慮する評価尺度である損失値の良し悪しに相関が見られる．損失関数を用いない $\alpha = 0$ の場合に比べて，$\alpha = 0.1$ では精度，F 値，損失値とも向上した．一方，0.1 より大きい α の値ではすべての評価尺度で $\alpha = 0$ を下回った．$\alpha > 0$ では α の値が大きくなるにつれて 3 つの評価尺度とも低下する傾向があり，式 (3) における内積に対する損失関数の比率が大きすぎると，重みの学習を阻害しているものと思われる．今後の課題として，損失関数が学習に与える影響をさらに調べる必要がある．

表 1: 実験結果

α	精度	$F_{\beta=1}$	損失値
0	80.80	52.81	0.905
0.1	81.06	54.13	0.875
0.2	80.68	52.23	0.959
0.3	79.00	47.81	1.035
0.4	76.92	43.68	1.317
0.5	74.52	42.77	1.294

参考文献

[1] Nadeau, David and Sekine, Satoshi. A survey of named entity recognition and classification, *Lingvisticae Investigationes*, Vol. 30, No. 1, pp. 3–26, 2007.

[2] Collins, Michael. Discriminative training methods for hidden Markov models: theory and experiments with perceptron algorithms, *EMNLP–2002*, pp. 1–8, 2002.

[3] Dekel, Ofer and Keshet, Joseph and Singer, Yoram. Large margin hierarchical classification, *ICML–2004*, pp. 209–216, 2004.