先読みを用いた単語系列ラベリングへの最易優先方策の適用

佐野 峻平 †　三輪 誠 ‡　鶴岡 慶雅 †　近山 隆 †

† 東京大学

‡ マンチェスター大学

{sano|tsuruoka|chikayama}@logos.t.u-tokyo.ac.jp, makoto.miwa@manchester.ac.uk

1　はじめに

自然言語処理における重要なタスクの一つに, 品詞タグ付け・チャンキング・固有表現認識などの, 文中の各単語に正しいタグ付けを行うことを目的とする系列ラベリングがある. 学習データも計算機の計算能力も十分あると仮定した場合, 系列ラベリングで最高精度を出すためには文中の各語につき, 全てのラベルの組合せの任意の素性を考えてやればよい. しかし任意の素性を考えるためには学習データも計算能力もあまりに乏しいのが現状であり, この様な仮定は現実的ではない. そこで, 限られた学習データと計算能力の下でも高い精度を出せるよう様々な工夫がされてきた.

工夫の一つとして, 一つ一つの語の分類問題に分割し順次一つずつ行うことで現実的な計算時間で系列ラベリングを行う, 履歴に基づいたモデル　(history-based models) がある [6, 7, 8, 11]. 履歴に基づいたモデルは実装が容易, 学習効率が良いなどの利点があるが, 精度で勝る条件付確率場 (Conditional Random Fields, CRFs) [3] の登場によりあまり使用されなくなってきていた. しかし, 履歴に基づいたモデルを改良させた先読みによる手法 [9] は, CRFs と同等の計算量でより高い精度を出しており, CRFs では難しい, 素性の柔軟な設計も出来るという利点がある.

履歴に基づいたモデルでは, 探索順序を考慮する必要がある. 探索順序として, ラベル付けを確実性の高い語から行う最易優先方策 [2, 10] の有効性が示されている. この方策は, 人が系列ラベリングを行う場合は端から順ではなく確実な順に行うことからコンピュータにおいてもその順の探索が有効ではないか, という考えに基づき提案された. 先読みを含む従来の履歴に基づいたモデルでは, 「前から後ろ」や「後ろから前」のように端から順の探索が用いられてきたが, 最易優先方策は端から順の探索よりも精度が高い.

本研究では, 端から順より精度が高い事が示された, 最易優先方策を先読みによる手法に適用し, 先読みによる手法の改善を図ることを目的とする. 端から順の先読みと比べ, 提案手法では探索順序に工夫がなされているため, より正確なモデルが作られることが期待される.

2　関連研究

2.1　先読み

先読みは履歴に基づいたモデルを拡張させた手法で, 系列ラベリングや係り受け解析 (Dependency parsing) などの構造予測問題において有効性が示されている [9]. 先読みとは, 状態の決定の際に, 対象となる語より後に状態の決定される語についても様々な状態の組合せを考え, それらにより実現される構造の評価を行うことで最良の状態を選択する過程の事を言う. 言い換えると, 状態決定のされていない状態空間を探索する過程の事を先読みという.

他の語の状態の情報というのは, 状態の決定の際に有用である. 例えば系列ラベリングでは, 既にラベル付けされた語のラベル情報を利用している履歴に基づいたモデルが, 他の語のラベル付け結果とは独立に一語ずつラベル付けをする手法よりも精度が高い [6]. 用いるラベル情報は間違っている可能性があるにもかかわらずその曖昧な情報を利用した場合の方が精度が高いという結果は, それだけラベル情報が重要だということを示している.

履歴に基づいたモデルではいわば過去に決定された状態の情報を利用しているといえる. 視点を変えると, 履歴に基づいたモデルではまだ状態の決定のされていない語の情報, つまり未来に決定される状態の情報は利用できないともいえる. 履歴に基づいたモデルでは利用できない, 未来に決定される状態の情報も利用しよう, というのが先読みの基本的な考えである.

2.1.1 先読みを用いた学習

入力列 $\boldsymbol{x} = (x_1, ..., x_T)$, 出力ラベル列 $\boldsymbol{y} = (y_1, ..., y_T)$ とする. 単語 x_i にラベル y_i を対応させる, 行動 $a \in \boldsymbol{A}$ を適切に選択していくことが系列ラベリングの目的となる. 次の行動 a は, $\boldsymbol{x}$, およびそれまでになされた行動の情報を利用して決定される. 先読みを用いた場合と用いなかった場合の学習の違いをパーセプトロンを例として示す. Collins (2002) によるパーセプトロンでは, 素性ベクトル $\Phi(S)$, 重み $\boldsymbol{w}$, 現在の状態 S_0 としたとき

$$a_m = \arg\max_{a \in \boldsymbol{A}} \Phi(\mathrm{U}(S_0, a)) \cdot \boldsymbol{w} \quad (1)$$

となる行動 a_m の選択を繰り返していく. 但し状態 S は, いくつかの単語とラベルとの対応がとられたある時点の事をいい, 状態 S から行動 a を行うことで遷移する状態を $\mathrm{U}(S, a)$ で表す. 単語とラベルの対応が正しい行動を a_c とすると, $a_m \neq a_c$ の場合に重みの更新を行う.

$$\boldsymbol{w} \leftarrow \boldsymbol{w} + \Phi(\mathrm{U}(S_0, a_c)) - \Phi(\mathrm{U}(S_0, a_m)) \quad (2)$$

深さ d の先読みでは, 更に d 回先までの全ての行動の組み合わせを考え, 行動後の状態のうち最も $\Phi(S) \cdot \boldsymbol{w}$ が高くなるような行動 a_m^0 を選択する.

$$(a_m^0, ..., a_m^d) = \arg\max_{a_0, .., a_d \in \boldsymbol{A}} \Phi(\mathrm{U}(S_0, a_0, ..., a_d)) \cdot \boldsymbol{w} \quad (3)$$

$\mathrm{U}(S_0, a_0, ..., a_d)$ は S_0 から順に行動 $a_0, ..., a_d$ を行ったときの状態とする. $a_m^0 \neq a_c$ のとき重みの更新を行う.

$$\boldsymbol{w} \leftarrow \boldsymbol{w} + \Phi(S_c^*) - \Phi(S_m^*) \quad (4)$$

但し,

$$S_c^* = \mathrm{U}(S_0, a_c, a_c^1, ..., a_c^d), S_m^* = \mathrm{U}(S_0, a_m^0, ..., a_m^d)$$
$$(a_c^1, ..., a_c^d) = \arg\max_{a_1, .., a_d \in \boldsymbol{A}} \Phi(\mathrm{U}(S_0, a_c, a_1, ..., a_d)) \cdot \boldsymbol{w}$$

とする.

2.2 最易優先方策

構造予測問題をコンピュータに解かせる場合, 従来手法では端から順に状態の決定を行っていた. しかし端から順に状態の決定を行う場合, 例えば始めの方に一つに確定しづらい状態がありその状態の決定を誤ってしまうと, 誤った状態の情報が後の状態の決定にも連鎖的に影響し次々と状態の決定を誤る恐れがある. そこで, 誤りの起こりにくい確実性の高い部分から状態の決定を行うのが最易優先方策である. この様に状態の決定を行っていく事で連鎖的に誤る可能性が減り, より正確な状態決定が可能となる. 自然言語処理においては, 系列ラベリングや係り受け解析でその有効性が示されている [2, 10].

Goldberg ら (2010) の最易優先方策を系列ラベリングに用いる場合, 探索アルゴリズムは以下のようになる.

1. 初期化. $\boldsymbol{x} = (x_1, ..., x_T)$, x_1, $\boldsymbol{X} = \{x_1, ..., x_T\}$
2. $\boldsymbol{x} \in \boldsymbol{X}$ についてスコアの計算. $\mathrm{Score}(\boldsymbol{x}) = max_{\boldsymbol{y}} \mathrm{score}(\boldsymbol{x}, \boldsymbol{y})$
3. $x_n = \arg\max_{\boldsymbol{x} \in \boldsymbol{X}} \mathrm{Score}(\boldsymbol{x})$ なる語 x_n のラベルを決定
4. $\boldsymbol{X} = \boldsymbol{X} - x_n$ とし $\boldsymbol{X} \neq \phi$ ならば 2 に戻る

尚, score は分類器により出力されるスコアとする. 計算コストは上記の通りに行うと $O(n^2)$ となるが, 2 でのスコアの計算はスコアの更新のあるもののみでよいので, 実際は $O(n \log n)$ となる [2].

3 先読みへの最易優先方策の適用

先読みと最易優先方策の組み合わせ方は幾通りも考えられるが, 本研究では, 学習の収束が保証できて, かつ現実的な時間で計算可能であり, 精度の向上が見込まれる二つの手法を提案する. 一つ目の手法は端から順の先読みと探索順序のみが異なり, 二つ目の手法は更に先読みする語も異なる. 順に探索アルゴリズムを説明する.

3.1 探索順序を変更した手法

この手法では, 探索順序が最易優先方策に従う点のみが端から順の先読みと異なる.

1. 事前に学習させた分類器を用い, $\boldsymbol{x} = (x_1, ..., x_T)$ を最易な順 $\boldsymbol{x'} = (x'_1, ..., x'_T)$ に並び替え. $\boldsymbol{x'}$ は, $i < j$ ならば $\mathrm{Score}(\mathrm{x}'_\mathrm{i}) \geq \mathrm{Score}(\mathrm{x}'_\mathrm{j})$ を満たす.
2. $\boldsymbol{x'}$ に対応する出力ラベル列 $\boldsymbol{y'} = (y'_1, ..., y'_T)$ を先読みを用いた分類器を用いて求める. 先読み深さ d で $x'_i \equiv x_j$ をラベル付けする場合, 先読みする単語は $x_{j+1}, ..., x_{j+d}$

1 と 2 で用いる分類器は異なっている. 1 ではパーセプトロンで学習させた, 他の語のラベル情報を用いず語毎に独立にラベル付けを行う分類器, 2 では式 (4) の更新式で重みを学習させた, 先読みを用いた分類器 [9] を使用する.

有効性が示されている最易優先方策を用いることで, 端から順の先読みよりも高い精度となることが期待される.

ラベルの種類 N とすると, 先読みの際にとる行動の組み合わせは, 端から順の先読みや二つ目の手法では N^{d+1} となる. この手法では, 先読みした単語に対応するラベルが既に決められている場合, とれる行動は一つなので, 組み合わせは $N^{d+1-\alpha}$ となる. そのため, 計算量が少なくて済むという利点もある.

3.2 先読みする語も変更した手法

この手法では先読みする語も異なる. 最易優先方策に従って決められたラベル付け順序での, ラベル付け対象とする語より後の語を先読みする.

1. 事前に学習させた分類器を用い, $\boldsymbol{x} = (x_1, ..., x_T)$ を最易な順 $\boldsymbol{x'} = (x'_1, ..., x'_T)$ に並び替え. $\boldsymbol{x'}$ は, $i < j$ ならば $\mathrm{Score(x'_i)} \geq \mathrm{Score(x'_j)}$ を満たす.
2. $\boldsymbol{x'}$ に対応する出力ラベル列 $\boldsymbol{y'} = (y'_1, ..., y'_T)$ を先読みを用いた分類器を用いて求める. 先読み深さ d で x'_i をラベル付けする場合, 先読みする単語は $x'_{i+1}, ..., x'_{i+d}$

一つ目の手法とは先読みする語のみが異なる. この手法では, 端から順の先読みと同様まだラベル付けのされていない語のラベル情報を利用しているため個々のラベル付けが端から順と同程度に正確で, 最易な順にラベル付けを行うためラベル情報の信頼性がより高い事が期待される. つまり, 先読みと最易優先方策の両方の利点を併せ持っていることが期待される.

4 評価

4.1 評価の設定と結果

評価は, 品詞のタグ付けを用いて端から順の先読みの手法 [9] と比較することで行う. 全ての分類器の学習には, マージンありの平均化パーセプトロン [1] を用いる. マージンありのパーセプトロンは, SVMs (support vector machines) のようなマージン最大化の手法のように, マージン無しの場合のパーセプトロンよりも学習したモデルの汎化誤差が小さいことが知られている [4]. 本研究は手法の比較が目的であるため, 精度が向上する可能性はあるが素性のチューニングは行わず, 鶴岡ら (2011) と同じ素性を使う. マージンの値 (m) や学習の繰り返し回数 ($iter$) は不明のため, これらのパラメータについてはこちらで設定し, 同じ値同士で比較をする.

手法	m	$iter$		
		1	4	7
Lookahead[9]	0	96.89	97.16	97.20
	40	96.87	97.22	97.24
	100	96.66	97.15	97.22
提案手法 1 (探索順序変更)	0	96.80	97.09	97.09
	40	96.82	97.16	97.21
	100	96.55	97.09	97.18
提案手法 2 (先読み語も変更)	0	96.79	97.04	n/a
	40	96.82	97.17	97.18
	100	96.47	97.03	n/a

表 1: 品詞のタグ付けの評価結果 (開発データ)

品詞のタグ付けの実験には, Penn Treebank[5] の Wall Street Journal (WSJ) コーパスを用い, section 0-18 を学習データ, section 19-21 を開発データとした. 探索順序の並び替えに用いる分類器は, $m = 40$, $iter = 7$ で学習させた. 端から順の先読み, 及び提案手法については, 先読み深さ $d = 1$, $m = 0, 40, 100$, $iter = 1, 4, 7$ とした. 結果を表 1 に示す.

4.2 考察

マージンと繰り返し回数が同じもの同士を比較すると, 端から順の先読みが最も精度が高く, 次いで探索順序のみ変更した手法, 最後に先読みする語も変更した手法という結果となり, 目的としていた先読みの手法の改善は達成出来なかった. この結果から, 先読みに於いては局所的な情報が精度の向上に大きく貢献していると考えられる. 端から順の先読みでは, ラベル付けの対象とする単語より前の単語は全て既にラベルが付けられている. また, 先読みする単語もラベル付けの対象とする語の後ろの語であり, ラベル情報を十分に利用できる. 探索順序のみ変更した手法では, 前の語のラベル情報は利用できないことがあり, 先読みする単語も変更した手法では後ろの語のラベル情報も利用できないことがある. 提案手法では最易な順にラベル付けしているため用いるラベル情報の信頼性は高

いが, それ以上に, 局所的な情報を多く利用できる手法の方が精度が高いという結果となった.

ただ, 先読みへ最易優先方策を適用させる方法はいくつもあるため, 先読みに最易優先方策を適用しても手法の改善は出来ないと結論付けられる訳ではない. 例えば, 探索順序を動的に変更することが考えられる. 本研究の提案手法では, 探索順序は初めに決めたもので固定していた。しかし Goldberg ら (2010) の最易優先方策では, ラベルを付ける度に動的に探索順序を変更しており, 提案手法でもそれを行うことで精度が向上する可能性がある.

探索順序の決定に用いた分類器についても, 改善の余地がある. 学習データはラベル付けを行う分類器と同じものを用いており, 過学習が起こっている可能性があるため, 学習データを分ける, 交差検定 (Cross-Validation) を行うなど, 過学習を防ぐ方法を試したい. また, パラメータを変えることで結果が変わる可能性もある.

また, 系列ラベリングの他のタスクでも評価を行い, この結果がタスクに依存しないかの調査も必要である.

5 おわりに

本研究では, 系列ラベリングを解く手法として, 先読みに最易優先方策を適用させた手法を提案した. 端から順の先読みの手法と比較して精度で上回ることは出来ず, 当初の目的であった手法の改善は達成出来なかった. 探索順序のみ変更した手法と, 先読みする単語も変更した手法では, 探索順序のみ変更した手法の方が高い精度を出した. 今後は, 動的に探索順序を変えていく手法の評価や諸パラメータのチューニングなどを行い, より良い手法の模索を行いたいと考えている.

参考文献

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

[2] Yoav Goldberg and Michael Elhadad. An efficient algorithm for easy-first non-directional dependency parsing. *NAACL-HLT*, pp. 742–750, 2010.

[3] John Lafferty, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML'01*, pp. 282–289, 2001.

[4] Yaoyong Li, Hugo Zaragoza, Ralf Herbrich, John Shawe Taylor, and Jaz S. Kandola. The perceptron algorithm with uneven margins. *ICML*, pp. 379–386, 2002.

[5] Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. Building a large annotated corpus of english: Penn treebank. *Computational Linguistics*, pp. 313–330, 1994.

[6] Andrew McCallum, Dayne Freitag, and Fernando Pereira. Maximum entropy markov models for information extraction and segmentation. *ICML*, pp. 591–598, 2000.

[7] Joakim Nivre. Memory-based dependency parsing. *CoNLL*, pp. 49–56, 2004.

[8] Adwait Ratnaparkhi. A maximum entropy model for part-of-speech tagging, 1996.

[9] Yoshimasa Tsuruoka, Yusuke Miyao, and Jun'ichi Kazama. Learning with lookahead : Can history-based models rival globally optimized models ? *CoNLL*, pp. 58–73, 2011.

[10] Yoshimasa Tsuruoka and Jun'ichi Tsujii. Bidirectional inference with the easiest-first strategy for tagging sequence data. *HLT/EMNLP*, pp. 467–474, 2005.

[11] Hiroyasu Yamada and Yuji Matsumoto. Statistical dependency analysis with support vector machines. *IWPT*, pp. 195–206, 2003.