Vol.2018-NL-236 No.8
2018/7/10

重み行列の対称性および巡回性の仮定に基づく双アフィン分類器の冗長性削減

松野 智紀[1,a)]　林 克彦[2]　石原 敬大[1]　真鍋 陽俊[3]　松本 裕治[1]

概要：現在，二項関係のモデル化に注意機構を導入するための手法として双アフィン変換が大きな注目を集めている．例えば，係り受け解析の分野では Dozat and Manning (2017) によって提案された深層双アフィン構文解析器 (Deep Biaffine parser) が English Penn Treebank や CoNLL 2017 shared task でグラフ型構文解析器としての最高精度を達成した．一方で，双アフィン変換の重み行列 (双線型変換項) は n^2 個の過剰なパラメータを持つため (n は次元数)，学習データが少ない場合などにモデルが過学習することが報告されている．本稿では深層双アフィン構文解析器における双アフィン変換の重み行列に対称性あるいは巡回性という仮定を加えることで，その冗長性の削減を試みた．CoNLL 2017 shared task のデータセットを使った実験から，重み行列に巡回行列を用いたモデルはシステム全体のパラメータ数を約 18%削減できる上，ほとんどの言語でベースラインと同程度かそれ以上の解析精度を達成できることがわかった．

1. はじめに

近年，自然言語処理の様々なタスクで注意機構に基づく手法が用いられている [1, 16]．係り受け解析などの 2 つの単語を項とする**二項関係**を扱うタスクにおいても多くのモデルが注意機構を導入することで高い性能を達成してきた [11, 6, 4]．

二項関係に注意機構を導入するための手法の一つとして**双アフィン変換**に基づく手法がある (ここでは文献 [4] に従って，この手法を**双アフィン分類器**と呼ぶ．)． Dozat and Manning [4] は係り受け解析における係り受けスコアの計算に双アフィン分類器を用いることで，English Penn Treebank におけるグラフ型係り受け解析器の最高精度を達成している (**深層双アフィン構文解析器**)．また，遷移型構文解析器においても，Ma et al [14] がある時点から次にスタックへ入力される単語の確率分布として双アフィン分類器を用い，English Penn Treebank での最高精度を記録している．

二項関係のモデル化において，双アフィン変換は非常に豊かな表現力を有する一方で，その重み行列 (双線型変換項) が $O(n^2)$ という過剰なパラメータを持つことになる (n は次元数)． そのため，メモリが豊富でない GPU や携帯端末などの環境下でモデルを運用する場合，このことはシステム実用化の妨げになる可能性がある．さらに，学習データが少量の場合，その過剰なパラメータ数からモデルの自由度が高くなり，過学習を引き起こす原因となることも懸念される [15]．

本稿では双アフィン分類器に用いる重み行列に対称性または巡回性の仮定を設けることで，その冗長性の削減を試みる．対称性及び巡回性の仮定は行列をベクトル化することが可能となり，空間計算量を $O(n)$ に削減することができる． また，スコア計算 1 回当たりの時間計算量は，対称性の場合，$O(n)$，巡回性の場合，高速フーリエ変換を用いて $O(n \log n)$ となる．さらに，対称性に基づくモデルの表現力は制限される一方で[*1]，巡回性に基づくモデルは元の双アフィン分類器と本質的に (次元数の議論を除いて) 等価な表現力を保持する．

実験では深層双アフィン構文解析器における双アフィン分類器の重み行列に制限を加え，その解析性能への影響を分析した．実験データには CoNLL 2017 shared task の係り受け解析データセットを採用し，その中から学習データ量が比較的豊富な 5 言語，少量の 4 言語の計 9 言語を採用した． 実験結果からは提案手法が特に学習データが少量の言語において有効であることが確認された．

[1] 奈良先端科学技術大学院大学
[2] 大阪大学 産業科学研究所
[3] ワークスアプリケーションズ
[a)] matsuno.tomoki.mr1@is.naist.jp

[*1] ある単語 a, b が係り受け関係を持つ場合，双線型変換項のスコアを $f(a, b)$ とする．このとき，双線型変換項が対称性を持つ場合，$f(a, b) = f(b, a)$ となるため，係り受け関係のような有向辺を表すのには不向きとなる．

2. 深層双アフィン構文解析器

本稿で扱うモデルは Dozat and Manning [4] によって提案された深層双アフィン構文解析器を基本とする．このモデルの構成は CoNLL 2017 shared task on Universal Dependency Parsing で最高精度を達成したものである [20]．この構成において最も重要な要素はスコア計算部分である．

このモデルでは，入力として単語/品詞列を受け取り，各単語/品詞間で，係り受け関係 (弧)，及び，その文法機能ラベル付与に対する予測スコアを計算する．スコア計算には LSTM，多層パーセプトロン (MLP)，及び，双アフィン分類器が用いられる．

以下では，まず双アフィン分類器の主幹となる双アフィン変換について説明を行うが，簡略化のため，LSTM と MLP については説明を省略する．次に，モデルの全体構成について概略を示す．

2.1 双アフィン変換

係り受け解析のスコア関数は二項関係をモデル化するため，以下のような双アフィン変換

$$g(\mathbf{v}_i, \mathbf{v}_j) = \mathbf{v}_i^{\mathrm{T}} \mathbf{A} \mathbf{v}_j + (\mathbf{v}_i \oplus \mathbf{v}_j)^{\mathrm{T}} \mathbf{b} + b \qquad (1)$$

を用いる．ここで $\oplus$ はベクトルの結合を表す．式 (1) の右辺第 1 項 (双線型変換項) で $\mathbf{v}_i$ と $\mathbf{v}_j$ の関連度スコアを，第 2 項で $\mathbf{v}_i$ および $\mathbf{v}_j$ がそれぞれ独立に生起するスコアを表すことができる． b はバイアスとする．

2.2 モデルの構成

このモデルは以下の構成になっている．

(1) まず，入力文の単語列と品詞列から各単語のベクトル表現を作り，3 層の双方向 LSTM によって新たな単語表現にエンコードする．ここでは，i 番目の単語 w_i に対応する双方向の出力を結合したものを $\mathbf{y}_i$ で示す．

(2) 次に MLP を用いて LSTM の出力を変換する．ここでは弧とラベルの予測それぞれにおいて係り先と係り元で異なる MLP を使う．

$$\begin{aligned} \mathbf{v}_i^{arc-head} &= \mathrm{MLP}^{arc-head}(\mathbf{y}_i), \\ \mathbf{v}_j^{arc-dep} &= \mathrm{MLP}^{arc-dep}(\mathbf{y}_j), \\ \mathbf{v}_i^{label-head} &= \mathrm{MLP}^{label-head}(\mathbf{y}_i), \\ \mathbf{v}_j^{label-dep} &= \mathrm{MLP}^{label-dep}(\mathbf{y}_j). \end{aligned} \qquad (2)$$

ここで弧に対するベクトルの次元数は n，ラベルに対するベクトルの次元数は m とする．

(3) 双アフィン変換を用いて各単語ペア間の係り受けのスコア $s_{i,j}^{(arc)}$ を計算する．

$$s_{i,j}^{(arc)} = \mathbf{v}_i^{arc-head\mathrm{T}} \mathbf{W} \mathbf{v}_j^{arc-dep} + \mathbf{v}_i^{arc-head\mathrm{T}} \mathbf{b}^{(arc)}. \qquad (3)$$

式 (3) の第 1 項は単語ペア (w_i, w_j) のかかりやすさを，第 2 項は単語 w_i の主辞になりやすさを表現している．

(4) 式 (3) で求めた全単語ペアのスコアを入力として Chu-Liu/Edmonds アルゴリズムを走らせ，合計スコアが最大となるような木構造を得る．

(5) 各単語 w_j について，その予測された係り先 w_i との弧にラベル l を付与するスコア $s_{i,j}^{(l)}$ ($l \in \{1, 2, ..., L\}$; L:ラベルの数) を全てのラベルに対して計算する．計算式は以下で定義される．

$$\begin{aligned} s_{i,j}^{(l)} = {} & \mathbf{v}_i^{label-head\mathrm{T}} \mathbf{U}^{[l]} \mathbf{v}_j^{label-dep} \\ & + (\mathbf{v}_i^{label-head} \oplus \mathbf{v}_j^{label-dep})^{\mathrm{T}} \mathbf{b}^{[l]} \\ & + b^{[l]}. \end{aligned} \qquad (4)$$

ここでは，それぞれのラベル毎に異なる重み行列 $\mathbf{U}^{[l]}$，重みベクトル $\mathbf{u}^{[l]}$，バイアス $b^{[l]}$ を用いる．式 (4) の右辺第 1 項は係り先 w_i と係り元 w_j との間に張る弧にラベル l を付与するスコアを表現している．第 2 項は係り先及び係り元がそれぞれ単独で与えられたときのラベルのスコアを表している．

本稿における実験では，深層双アフィン構文解析器におけるパラメータの約 19%が $\mathbf{W} \in \mathbb{R}^{n \times n}$ と $\mathbf{U}^{[l]} \in \mathbb{R}^{m \times m}$ によるものであった．これらのパラメータ削減はシステムのメモリ効率向上のみならず，モデル学習における過学習の緩和も期待される．

3. 提案手法

本稿では，式 (3) の係り受けスコア関数における重み行列 $\mathbf{W}$ と式 (4) のラベル付与スコア関数における重み行列 $\mathbf{U}^{[l]}$ ($\forall l \in \{1, 2, \ldots, L\}$) に対称性あるいは巡回性の仮定を設けることでパラメータ数の削減を行う．

3.1 提案手法 1 ：対称行列を用いたスコア関数

ここでは双線型変換項の重み行列が**対称行列**であると仮定し対角化を行う．結果として，スコア関数の双線型変換項を 2 つの入力ベクトルと重みベクトルの三重内積に変形することができる．

3.1.1 重み行列の対角化による双線型変換項の変形

$\mathbf{W} \in \mathbb{R}^{n \times n}$ が対称行列のとき，以下のように直交行列 $\mathbf{O} \in \mathbb{R}^{n \times n}$ を用いて $\mathbf{W}$ を対角化可能である:

$$\mathbf{W} = \mathbf{O} \mathrm{diag}(\mathbf{w}) \mathbf{O}^{\mathrm{T}}.$$

ここで，$\mathbf{w} \in \mathbb{R}^n$ は行列 $\mathbf{W}$ の固有値を並べたベクトルであり，$\mathrm{diag}(\mathbf{w})$ はベクトル $\mathbf{w}$ を対角成分とする対角行列を表す．これを用いて，双線型変換項を以下のように変形することができる:

$$\begin{aligned}\mathbf{v}_i^{\mathrm{T}}\mathbf{W}\mathbf{v}_j &= \mathbf{v}_i^{\mathrm{T}}\mathbf{O}\,\mathrm{diag}(\mathbf{w})\mathbf{O}^{\mathrm{T}}\mathbf{v}_j \\ &= \mathbf{v}_i'^{\mathrm{T}}\,\mathrm{diag}(\mathbf{w})\mathbf{v}_j' \\ &= \langle \mathbf{v}_i', \mathbf{w}, \mathbf{v}_j' \rangle.\end{aligned} \tag{5}$$

ここで $\mathbf{v}_i' = \mathbf{O}^{\mathrm{T}}\mathbf{v}_i$, $\mathbf{v}_j' = \mathbf{O}^{\mathrm{T}}\mathbf{v}_j$ であり，$\langle \mathbf{v}_i', \mathbf{w}, \mathbf{v}_j' \rangle$ は $\mathbf{v}_i', \mathbf{w}$ および $\mathbf{v}_j'$ による 3 重内積 ($\langle \mathbf{a}, \mathbf{b}, \mathbf{c} \rangle = \sum_{k=1}^{n} a_k b_k c_k$) である．結果として，対称行列の制約により行列のパラメータ数は n^2 から n に削減される．

3.1.2 同時対角化

対称行列の組が可換族をなすとき，共通の直交行列を用いて対角化を行うことができる [12]．そこで，提案手法 1 において L 個のラベル付与スコア関数の重み行列 $\mathbf{U}^{[1]}, \mathbf{U}^{[2]}, \ldots, \mathbf{U}^{[L]}$ が可換族を成すと仮定する．すなわち，

$$\mathbf{U}^{[p]}\mathbf{U}^{[q]} = \mathbf{U}^{[q]}\mathbf{U}^{[p]},\ \forall p, q \in \{1, 2, \ldots, L\}$$

であるとする．これにより，L 個の重み行列すべてを同時に対角化することが可能となる．したがって，式 (5) の変形において，$\mathbf{v}_i^{label-head}$ および $\mathbf{v}_j^{label-dep}$ がどのラベル付与スコア関数についても共通の直交行列で写像されることが保証される．

3.1.3 スコア関数

以上を踏まえて対称行列の仮定のもと，双アフィン変換の双線型変換項を 3 重内積で置き換える．まず弧に対するスコア関数は以下のように定義される:

$$\begin{aligned}s_{i,j}^{(arc)} &= \langle \mathbf{v}_i^{arc-head}, \mathbf{w}, \mathbf{v}_j^{arc-dep} \rangle \\ &\quad + (\mathbf{v}_i^{arc-head} \oplus \mathbf{v}_j^{arc-dep})^{\mathrm{T}}\mathbf{b}.\end{aligned} \tag{6}$$

ここで $\mathbf{w} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^{2n}$ である．式 (6) には式 (3) と異なり第 2 項に係り元のベクトルも含まれているが，これは実験により係り元と係り先両方のベクトルを含めたほうが性能が向上することが確認されたからである．

また，ラベル l の付与に対するスコア関数は以下のように定義される．

$$\begin{aligned}s_{i,j}^{(l)} &= \langle \mathbf{v}_i^{label-head}, \mathbf{u}^{[l]}, \mathbf{v}_j^{label-dep} \rangle \\ &\quad + (\mathbf{v}_i^{label-head} \oplus \mathbf{v}_j^{label-dep})^{\mathrm{T}}\mathbf{b}^{[l]}.\end{aligned} \tag{7}$$

ここで $\mathbf{u}^{[l]} \in \mathbb{R}^m$, $\mathbf{b}^{[l]} \in \mathbb{R}^{2m}$ である．式 (4) における $b^{[l]}$ は式 (7) では省かれているが，これは $b^{[l]}$ の有無が性能にほどんど影響しないことを確認したためである．

3.2 提案手法２：巡回行列を用いたスコア関数

文献 [15] では，双線型変換の重み行列に巡回行列を用いることで効果的に行列のパラメータ数を削減し，計算効率を向上させる手法が提案されている．これと同様に，双アフィン変換の双線型変換項の重み行列を巡回行列とすることでパラメータ数を削減したスコア関数を提案する．

3.2.1 巡回行列による双線型変換

ベクトル $\mathbf{w} \in \mathbb{R}^n$ に対する巡回行列 $C(\mathbf{w}) \in \mathbb{R}^{n \times n}$ を以下のように定義する:

$$C(\mathbf{w}) = \begin{bmatrix} w_1 & w_n & \ldots & w_3 & w_2 \\ w_2 & w_1 & w_n & & w_3 \\ \vdots & w_2 & w_1 & \ddots & \vdots \\ w_{n-1} & & \ddots & \ddots & w_n \\ w_n & w_{n-1} & \ldots & w_2 & w_1 \end{bmatrix}. \tag{8}$$

ここで，$\mathbf{w}^{\mathrm{T}} = (w_1, \ldots, w_n)$ である．この巡回行列 $C(\mathbf{a})$ を用いた重み行列のパラメータ数が n の双線型変換を考えることができる:

$$\mathbf{v}_i^{\mathrm{T}} C(\mathbf{w}) \mathbf{v}_j. \tag{9}$$

3.2.2 スコア関数

式 (9) を双線型変換項に採用した双アフィン変換によるスコア関数を提案する．まず，弧に対するスコア関数は以下のように定義される:

$$\begin{aligned}s_{i,j}^{(arc)} &= \mathbf{v}_i^{arc-head} C(\mathbf{w}) \mathbf{v}_j^{arc-dep} \\ &\quad + (\mathbf{v}_i^{arc-head} \oplus \mathbf{v}_j^{arc-dep})^{\mathrm{T}}\mathbf{b}.\end{aligned} \tag{10}$$

ここで $\mathbf{w} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^{2n}$ である．つづいて，ラベル付与に対するスコア関数は以下のように定義される:

$$\begin{aligned}s_{i,j}^{(l)} &= \mathbf{v}_i^{label-head} C(\mathbf{u}^{[l]}) \mathbf{v}_j^{label-dep} \\ &\quad + (\mathbf{v}_i^{label-head} \oplus \mathbf{v}_j^{label-dep})^{\mathrm{T}}\mathbf{b}^{[l]}.\end{aligned} \tag{11}$$

ここで $\mathbf{u}^{[l]} \in \mathbb{R}^m$, $\mathbf{b}^{[l]} \in \mathbb{R}^{2m}$ である．

3.2.3 高速フーリエ変換による計算の効率化

ここでは，式 (9) の計算を高速フーリエ変換を用いて効率化するための方法を説明する．n 次離散フーリエ変換の行列表現を $\mathfrak{F}_n \in \mathbb{C}^{n \times n}$ とすると，任意の巡回行列 $C(\mathbf{w}) \in \mathbb{R}^{n \times n}$ は以下のように対角化できることが知られている [5]:

$$C(\mathbf{w}) = \mathfrak{F}_n^{-1}\,\mathrm{diag}(\mathfrak{F}_n\mathbf{w})\mathfrak{F}_n.$$

これを用いて式 (9) を複素 3 重内積に展開することができる [12]:

$$\begin{aligned}\mathbf{v}_i^{\mathrm{T}} C(\mathbf{w}) \mathbf{v}_j &= \mathbf{v}_i \mathfrak{F}_n^{-1}\,\mathrm{diag}(\mathfrak{F}_n\mathbf{w})\mathfrak{F}_n\mathbf{v}_j \\ &= \frac{1}{n}\overline{\mathfrak{F}_n\mathbf{v}_i}^{\mathrm{T}}\,\mathrm{diag}(\mathfrak{F}_n\mathbf{w})\mathfrak{F}_n\mathbf{v}_j \\ &= \langle \mathbf{v}_i', \mathbf{w}', \mathbf{v}_j' \rangle \\ &= \Re(\langle \mathbf{v}_i', \mathbf{w}', \mathbf{v}_j' \rangle)\end{aligned} \tag{12}$$

ここで，$\mathbf{v}_i' = \overline{\mathfrak{F}_n\mathbf{v}_i}$, $\mathbf{v}_j' = \mathfrak{F}_n\mathbf{v}_j$, $\mathbf{w}' = \frac{1}{n}\,\mathrm{diag}(\mathfrak{F}_n\mathbf{w})$ であり，それぞれ n 次元複素ベクトルとなっている．また，$\Re(\cdot)$ は実部をとる操作である．複素三重内積に展開した結果として，巡回行列による双線型変換は高速フーリエ変換 (FFT) により $O(n \log n)$ で計算することができる．

実際の実験では，$\mathbf{w}'$ について実数ベクトルの離散フーリエ変換で初期化したのち，学習時の値更新は複素空間で行った．実数ベクトルを離散フーリエ変換して得られる複素ベクトルは共役対称性を持つが，学習中にこの性質を保持するための議論は文献 [21, 7] で議論されている．この議論は勾配学習計算で用いられる複素計算が実数空間での計算に対応がとれていれば，共役対称性が保持されるというものである．本稿のモデル学習も時間空間と周波数空間で対応がとれた演算 (積，和，内積) のみを利用するので，初期化時点で共役対称性が保たれていれば，共役対称性は学習後も保持される[*2]．

3.2.4 巡回行列による双線型変換の表現力

ここでは任意の正方行列 $\mathbf{W} \in \mathbb{R}^{n \times n}$ に対して，巡回行列の表現力について考える．文献 [19] では $\mathbf{W}$ に対して，$\mathbf{W} = \Re(\mathbf{W}')$ となる正規行列 $\mathbf{W}' \in \mathbb{C}^{n \times n}$ が必ず存在することが証明されている．対称行列と同様に，正規行列は以下のように対角化できる．

$$\mathbf{W} = \Re(\mathbf{W}') = \Re(\mathbf{O}\,\mathrm{diag}(\mathbf{w}')\mathbf{O}^*).$$

ここで $\mathbf{O} \in \mathbb{C}^{n \times n}$ はユニタリ行列，$\mathbf{O}^*$ は $\mathbf{O}$ の共役転置であり，$\mathbf{w}' \in \mathbb{C}^n$ は固有値を表す複素ベクトルである．ここで上式を双線型変換項に導入した形は式 (12) に変形できる．ユニタリ行列 $\mathbf{O}$ は全単射であるので，双線型変換への入力ベクトル $\mathbf{v}_i$，$\mathbf{v}_j$ に関してはそれらと一対一に対応した点 $\mathbf{v}'_i = \mathbf{O}^{\mathrm{T}}\mathbf{v}_i$, $\mathbf{v}'_j = \mathbf{O}^{\mathrm{T}}\mathbf{v}_j$ を学習すると考えられる．以上より，提案手法 2 のモデルは元の双アフィン分類器と等価な表現力を有する．ただし，巡回行列では，前述した共役対称性の議論があるため，理論上は $2n$ 次元の $\mathbf{w}$ を定義したとき，$\mathbf{W} \in \mathbb{R}^{n \times n}$ による双線形変換と等価な表現力を持つ．また，式 (12) への双線形変換項の変形のために，ラベル付与スコア関数の重み行列の組 $\mathbf{U}^{[1]}, \mathbf{U}^{[2]}, \ldots, \mathbf{U}^{[L]}$ をそれらを実部とする正規行列の組 $\mathbf{U}'^{[1]}, \mathbf{U}'^{[2]}, \ldots, \mathbf{U}'^{[L]}$ で表現し，その同時対角化を考える際には，3.1.2 で述べた議論と同様に，それらが可換族を成すという制約が必要である．

4. 関連研究

係り受け解析

近年，注意機構を用いたグラフ型構文解析器が多く提案されている．文献 [11] は機械翻訳で用いられている注意機構 [1] をグラフ型構文解析器に組み入れた．彼らのモデルは，文中のそれぞれの単語に対する双方向 LSTM の出力を係り元の候補と結合したものを MLP に入力し係り受け関係を予測する．同様に，文献 [6] はマルチタスクニューラルモデルの中のグラフ型構文解析器で 文献 [11] の MLP 分類器を双線型変換分類器に置き換えたものを提案した．続いて文献 [4] は 文献 [11] の手法に変更を加え，MLP 分類器の代わりに双アフィン変換を用いることで，単語ペア間の係りやすさに加えてある単語が主辞となるような事前確率を表現した．遷移型構文解析器においても，現在 English Penn TreeBank で最高精度を持つ 文献 [14] は，ある時点から次にスタックへの入力として選ばれる単語の確率分布として入力文中の各単語に対応する LSTM 出力を用いた双アフィン変換による注意機構を用いている．本稿で提案した手法はこれらのモデルにも統合可能である．

ニューラルネットにおけるパラメータ削減

近年，ニューラルネットのパラメータ削減を行う手法が数多く提案されている．

提案手法に類似するアプローチとして，低ランク近似によって写像行列をより小さな行列へと分解する方法がある [13]．また，Ishihara et al. [10] はニューラルテンソルネット [18] に対して，対称行列及び正規行列の固有値分解を導入し，パラメータ削減の影響を分析している．双線型変換項に対するパラメータ削減を考えている点において，本稿は文献 [10] に近いが，ここでは深層双アフィン構文解析器に対してモデル化を行っている点で異なる．

写像行列のパラメータを何らかの形で共有してその数を削減する方法がある．文献 [3] では畳み込みニューラルネットにおける全結合層の行列を巡回行列としているが，本稿では双線型変換項に巡回行列を用いる点で異なる．文献 [2] ではハッシュカーネル，文献 [17] ではテプリッツ行列のような特殊な行列を利用してニューラルネットのパラメータ削減を実現している．これらは双線型変換項に利用することも可能であるが，実対角行列，巡回行列に基づく手法の方が計算効率は優れる．

文献 [8] では蒸留 (distillation) と呼ばれるモデル再学習法を提案し，元のモデルよりもコンパクトなモデルを学習することに成功している．しかし，モデルの再学習を行うため，学習に多くの時間を要する． 文献 [9] では量子化によって大幅なパラメータ削減を実現しているが，精度は元のモデルよりも劣ることが報告されている．また，これらの手法は提案手法と併用することも原理的には可能である．

5. 実験

5.1 データセットと実装

上で述べたモデルを CoNLL 2017 shared task on Universal Dependency Parsing データセットのうち UD_Chinese, UD_Czech, UD_English, UD_German, 及び, UD_Ukrainian の 5 つの大規模なツリーバンクと UD_French-ParTUT, UD_Galician-TreeGal, UD_Latin, 及び, UD_Slovenian-SST の小規模なツリーバンクで比較した．比較モデルとして，同 shared task で最高精度を記録した Timothy Dozat に

[*2] 学習時の勾配ベクトルに対して，各次元ごとに異なるスケーリングを用いる場合，共役対称性は保たれなくなる．

よる構文解析器[*3]を使用した．提案モデルの実装は，比較モデルの実装を基礎とし，スコア関数のみを変更した．他は比較モデルと同一とした．次元数や学習率などのハイパーパラメータは，バケッティングのためのバケット数を変更したことを除いて，比較モデルのデフォルトの設定に従った．実験には正解の単語分割と正解の品詞を用いる．同 shared task は単語分割と品詞タグ付けを含むが，本研究の目的は提案手法の双アフィン分類器における影響を確かめることであるため，単語分割と品詞タグ付けはタスクから除外した．

5.2 結果

表 1，2 に解析結果をまとめた．表 1 は学習データが大量な言語の結果である．UD_German の UAS で対称行列に基づく手法がベースラインを上回った．また，UD_Czech において巡回行列に基づく手法がベースラインと比べて UAS で 0.01 ポイント，LAS で 0.04 ポイント差とベースラインに迫る性能を達成した．

表 2 は学習データが少量な言語での結果である．ここではすべての Treebank で巡回行列に基づく手法がベースラインを上回っている．これは冗長なパラメータを削減することで学習データが少量の場合において過学習に対する頑健性が増していることを示すものである．また，同手法はすべての Treebank で対称行列に基づく手法を上回っている．これは重み行列の巡回性の仮定によって単純な内積よりも係り元と係り先ベクトルの要素間の相互作用を豊かに表現できていることを示唆している．

5.3 分析

過学習緩和

訓練データ量がモデルに与える影響をさらに検証するため，UD_English で文例数を削減する実験を行った．表 3 にその結果を示す．この実験では対称行列に基づく手法は常にベースラインを下回っているが，巡回行列に基づく手法は文例数を半数以上削減した実験においてベースラインを上回った．これは巡回行列に基づく手法が，パラメータ削減によって過学習に対する頑健性を増しているだけでなく，少ない文例数からも高い汎化能力を獲得できていることを示唆している．

また，ベースラインの arc 分類器および label 分類器の次元数をそれぞれ 400 次元から 200 次元，100 次元から 50 次元に減らして，小規模なデータセットにおいて実験を行った．表 4 に示す結果から，単純にパラメータ数を削減する方法では，巡回行列に基づくモデル程の結果を得ることはできなかった．この結果は学習データが少ない場合における提案手法の有効性を示唆している．

[*3] https://github.com/tdozat/Parser-v2

パラメータ削減

表 5 はベースラインにおける各部位のパラメータ比率を表している．ただし，単語埋め込みおよび品詞埋め込みは含んでいない．ベースラインで総パラメータ数の最も大きな割合を占めているのは LSTM のパラメータで 67.22%だが，次に多いのは双アフィン分類器で，弧とラベルの分類器を合わせて 18.78%を占めている．表 6 は提案手法がベースラインと比べるとどちらも約 18%パラメータ数を削減したことを示している．これは双アフィン分類器のパラメータが大きく減ったことによる．

解析速度

解析速度の計測には NVIDIA 社の GPU である GTX 1080 を用いた．節 3 で述べたように，提案手法はどちらも理論上は時間計算量においてベースラインより優れるが，GPU を用いた実験ではベースラインが解析速度で提案手法を上回った．実際，UD_English の評価データの解析にかかった時間はベースラインが 14.99 秒に対し，提案手法は対称行列で 15.22 秒，巡回行列で 15.70 秒と同等な速度となった．

6. 結論

本稿では，双アフィン分類器の重み行列に対称性または巡回性の仮定を置くことでパラメータ削減を行い，その影響を CoNLL 2017 shared task on Universal Dependency Parsing データセットで検証した．結果，巡回行列に基づく手法は，モデルのパラメータ数を 18%以上削減しながら，学習データが少量な言語のすべてでベースラインを上回り，過学習に対する頑健性を示した．また，学習データが大量な言語においてもほとんどの言語で同等の精度を達成した．しかし，理論上は提案手法の時間計算量は比較手法のそれを下回っているはずであるにも関わらず，GPU 上での解析速度はほぼ同等であった．解析速度の向上は今後の課題である．

参考文献

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014. URL http://arxiv.org/abs/1409.0473.

[2] Wenlin Chen, James Wilson, Stephen Tyree, Kilian Weinberger, and Yixin Chen. Compressing neural networks with the hashing trick. In *International Conference on Machine Learning*, pages 2285–2294, 2015.

[3] Yu Cheng, Felix X. Yu, Rogério Schmidt Feris, Sanjiv Kumar, Alok N. Choudhary, and Shih-Fu Chang. An exploration of parameter redundancy in deep networks with circulant projections. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 2857–2865, 2015. DOI: 10.1109/ICCV.2015.327. URL

Treebank	ベースライン		対称行列		巡回行列	
	UAS	**LAS**	**UAS**	**LAS**	**UAS**	**LAS**
UD_Czech	**94.01**	**92.27**	93.79(-0.22)	92.01(-0.26)	94.00(-0.01)	92.23(-0.04)
UD_German	87.39	**84.48**	**87.58(+0.19)**	84.27(-0.21)	87.34(-0.05)	84.18(-0.30)
UD_English	**91.70**	**90.12**	91.42(-0.28)	89.72(-0.40)	91.24(-0.46)	89.48(-0.64)
UD_Ukrainian	**89.37**	**87.60**	89.13(-0.24)	87.27(-0.33)	89.28(-0.09)	87.37(-0.23)
UD_Chinese	**87.05**	**84.72**	86.85(-0.20)	84.33(-0.39)	86.78(-0.27)	84.44(-0.28)

表 1 大規模なツリーバンクでの結果.

Treebank	ベースライン		対称行列		巡回行列	
	UAS	**LAS**	**UAS**	**LAS**	**UAS**	**LAS**
UD_Slovenian-SST	74.57	68.60	75.53(+0.96)	69.14(+0.54)	**76.57(+2.00)**	**70.69(+2.09)**
UD_Latin	70.65	63.57	70.60(-0.05)	63.62(+0.05)	**71.35(+0.70)**	**64.75(+1.18)**
UD_French-ParTUT	92.05	90.51	91.82(-0.23)	89.90(-0.61)	**92.43(+0.38)**	**90.63(+0.12)**
UD_Galician-TreeGal	80.06	74.56	79.41(-0.65)	73.78(-0.78)	**80.72(+0.66)**	**75.89(+1.33)**

表 2 小規模なツリーバンクでの結果.

文例数の削減率	ベースライン		対称行列		巡回行列	
	UAS	**LAS**	**UAS**	**LAS**	**UAS**	**LAS**
0/4	**91.70**	**90.12**	91.42(-0.28)	89.72(-0.40)	91.24(-0.46)	89.48(-0.64)
1/4	**91.17**	**89.45**	90.80(-0.37)	89.03(-0.42)	90.91(-0.26)	89.15(-0.30)
2/4	90.20	88.19	89.74(-0.46)	87.74(-0.45)	**90.29(+0.09)**	**88.31(+0.12)**
3/4	88.45	86.13	88.36(-0.09)	86.08(-0.05)	**88.59(+0.14)**	**86.41(+0.28)**

表 3 訓練文例数を減らした UD_English での結果.

Treebank	UAS	LAS
UD_Slovenian-SST	74.70(-1.87)	68.78(-1.91)
UD_Latin	70.53(-0.82)	63.86(-0.89)
UD_French-ParTUT	91.78(-0.65)	89.86(-0.77)
UD_Galician-TreeGal	80.14(-0.58)	74.97(-0.92)

表 4 次元数を減らしたベースラインの結果. () 内は表 2 における巡回行列との差分を表す.

	次元	パラメータ数	比率
LSTM	200	1924800	67.22%
arc MLP	400	320800	11.20%
label MLP	100	80200	2.80%
arc 分類器	400	160400	5.60%
label 分類器	100	377437	13.18%
合計		2863637	100.00%

表 5 ベースラインにおけるパラメータ比率.

	ベースライン	対称行列	巡回行列
arc 分類器	160400	1200	1600
label 分類器	377437	11100	16400
共有部分との合計	2863637	2338100	2342200
ベースラインとの差分	0.0%	**-18.35%**	**-18.21%**

表 6 各モデルのパラメータ数比較.

https://doi.org/10.1109/ICCV.2015.327.

[4] Timothy Dozat and Christopher D. Manning. Deep biaffine attention for neural dependency parsing. *CoRR*, abs/1611.01734, 2016. URL http://arxiv.org/abs/1611.01734.

[5] Robert M Gray et al. Toeplitz and circulant matrices: A review. *Foundations and Trends® in Communications and Information Theory*, 2(3):155–239, 2006.

[6] Kazuma Hashimoto, caiming xiong, Yoshimasa Tsuruoka, and Richard Socher. A joint many-task model: Growing a neural network for multiple nlp tasks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1923–1933. Association for Computational Linguistics, 2017. URL http://aclweb.org/anthology/D17-1206.

[7] Katsuhiko Hayashi and Masashi Shimbo. On the equivalence of holographic and complex embeddings for link prediction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 554–559, 2017. DOI: 10.18653/v1/P17-2088. URL https://doi.org/10.18653/v1/P17-2088.

[8] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531, 2015.

[9] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Quantized neural networks: Training neural networks with low precision weights and activations constrained to +1 or -1. *CoRR*, abs/1609.07061, 2016. URL

http://arxiv.org/abs/1609.07061.

[10] Takahiro Ishihara, Katsuhiko Hayashi, Hitoshi Manabe, Masashi Shimbo, and Masaaki Nagata. Neural tensor networks with diagonal slice matrices. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 506–515, 2018. URL https://aclanthology.info/papers/N18-1047/n18-1047.

[11] Eliyahu Kiperwasser and Yoav Goldberg. Simple and accurate dependency parsing using bidirectional lstm feature representations. *Transactions of the Association for Computational Linguistics*, 4:313–327, 2016. URL http://aclweb.org/anthology/Q16-1023.

[12] Hanxiao Liu, Yuexin Wu, and Yiming Yang. Analogical inference for multi-relational embeddings. *CoRR*, abs/1705.02426, 2017. URL http://arxiv.org/abs/1705.02426.

[13] Zhiyun Lu, Vikas Sindhwani, and Tara N Sainath. Learning compact recurrent neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 5960–5964. IEEE, 2016.

[14] X. Ma, Z. Hu, J. Liu, N. Peng, G. Neubig, and E. Hovy. Stack-Pointer Networks for Dependency Parsing. *ArXiv e-prints*, May 2018.

[15] Maximilian Nickel, Lorenzo Rosasco, and Tomaso A. Poggio. Holographic embeddings of knowledge graphs. *CoRR*, abs/1510.04935, 2015. URL http://arxiv.org/abs/1510.04935.

[16] Alexander M Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685*, 2015.

[17] Vikas Sindhwani, Tara N. Sainath, and Sanjiv Kumar. Structured transforms for small-footprint deep learning. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 3088–3096, 2015. URL http://papers.nips.cc/paper/5869-structured-transforms-for-small-footprint-deep-learning.

[18] Richard Socher, Danqi Chen, Christopher D. Manning, and Andrew Y. Ng. Reasoning with neural tensor networks for knowledge base completion. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 926–934, 2013. URL http://papers.nips.cc/paper/5028-reasoning-with-neural-tensor-networks-for-knowledge-base-completion.

[19] Théo Trouillon, Christopher R. Dance, Éric Gaussier, Johannes Welbl, Sebastian Riedel, and Guillaume Bouchard. Knowledge graph completion via complex tensor factorization. *Journal of Machine Learning Research*, 18:130:1–130:38, 2017. URL http://jmlr.org/papers/v18/papers/v18/16-563.html.

[20] Daniel Zeman, Martin Popel, Milan Straka, Jan Hajic, Joakim Nivre, Filip Ginter, Juhani Luotolahti, Sampo Pyysalo, Slav Petrov, Martin Potthast, Francis Tyers, Elena Badmaeva, Memduh Gokirmak, Anna Nedoluzhko, Silvie Cinkova, Jan Hajic jr., Jaroslava Hlavacova, Václava Kettnerová, Zdenka Uresova, Jenna Kanerva, Stina Ojala, Anna Missilä, Christopher D. Manning, Sebastian Schuster, Siva Reddy, Dima Taji, Nizar Habash, Herman Leung, Marie-Catherine de Marneffe, Manuela Sanguinetti, Maria Simi, Hiroshi Kanayama, Valeria dePaiva, Kira Droganova, Héctor Martínez Alonso, Çağrı Çöltekin, Umut Sulubacak, Hans Uszkoreit, Vivien Macketanz, Aljoscha Burchardt, Kim Harris, Katrin Marheinecke, Georg Rehm, Tolga Kayadelen, Mohammed Attia, Ali Elkahky, Zhuoran Yu, Emily Pitler, Saran Lertpradit, Michael Mandl, Jesse Kirchner, Hector Fernandez Alcalde, Jana Strnadová, Esha Banerjee, Ruli Manurung, Antonio Stella, Atsuko Shimada, Sookyoung Kwak, Gustavo Mendonca, Tatiana Lando, Rattima Nitisaroj, and Josie Li. Conll 2017 shared task: Multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19. Association for Computational Linguistics, 2017. DOI: 10.18653/v1/K17-3001. URL http://www.aclweb.org/anthology/K17-3001.

[21] 林 克彦, 新保 仁, and 永田 昌明. フーリエ領域上でのホログラフィック埋め込み. In 言語処理学会第 *23* 回年次大会, pages 314–317, 2017.