双方向性と非線形性を考慮した上位語・下位語関係の推定

Estimation of Hypernym-Hyponym Relations Considering their Bi-directionality and Non-linearity

山根 丈亮[†] 高谷 智哉[‡] 山田 整[‡] 三輪 誠[†] 佐々木 裕[†]
Josuke Yamane Tomoya Takatani Hitoshi Yamada Makoto Miwa Yutaka Sasaki
[†]豊田工業大学 Toyota Technological Institute
[‡]トヨタ自動車株式会社 Toyota Motor Corporation
[†]{sd16432, makoto-miwa, yutaka.sasaki}@toyota-ti.ac.jp
[‡]{tomoya_takatani, hitoshi_yamada_aa}@mail.toyota.co.jp

1 はじめに

上位語とは「対象の単語よりもより一般的な意味を持つ単語」のことであり，質問応答システムや検索システム，オントロジーの構築など多くの自然言語処理のタスクにおいて重要な語彙関係である．

上位語・下位語関係の推定においては，単語を低次元（数百次元）の実数値ベクトルで表現する単語の分散表現 [1] が高い性能を達成している．しかしながら多くの上位語・下位語推定のモデル [2, 3] は上位語を推定することに焦点を絞っており，上位語と下位語を同時に推定することが出来るモデルは少ない．ある単語の上位語の下位語には元の単語が含まれるという上位・下位語関係の双方向性を考慮するためには，これらを同時に推定することが不可欠である．また，実数値ベクトル間の関係については，多くは線形の関係としてモデル化されている．

本研究では上位語・下位語関係の双方向性・非線形性を考慮するために，いくつかの言語処理のタスクで良い性能を示している深層学習に基づく正準相関分析 (Deep Canonical Correlation Analysis; DCCA) [4, 5, 6] を利用した，上位語・下位語関係の推定手法の確立をめざす．

2 関連研究

2.1 分類器による上位語･下位語関係の推定

サポートベクターマシン (SVM) などの分類器を用いて，与えられた 2 単語が上位語・下位語関係にあるか否かを 2 値で分類する手法が多く研究されている．与えられた 2 単語の分散表現をそれぞれ x, y とすると，分類器の入力となる特徴ベクトルは結合 $< x, y >$ や差分 $< x - y >$ などとして人手で設計される [7, 3]．これらの手法においては x と y の間の関係を学習するのではなく，y がどれだけ上位語として出現するかを学習してしまう “lexical memorization” という問題が発生することが知られている．この問題に対処するために，$< x, y >$ を特徴ベクトルとして学習した分類器を再学習する手法 [8] が提案されており，このタスクにおいて良い性能を示している．しかしながら，これらの手法には特徴ベクトルの設計を人手で行わなければならない．

2.2 線形写像に基づく上位語推定

任意の単語の分散表現 x をその上位語 y に写像する行列 Φ を最適化する手法がいくつか提案されている [2, 3]．Yamane ら [2] は上位語の分散表現 x を行列 Φ で写像したベクトルと下位語の分散表現 y が近くなるように，以下の目的関数を上位・下位語ペアの集合 C について最大化することで，Φ を最適化する手法を提案した．

$$J = \sum_{(x,y)\in C} \Big(\log \mathrm{sim}(x,\ y) + \sum_{y' \sim V}^{m} \log\left(1 - \mathrm{sim}(x,\ y')\right) \Big), \quad (1)$$

$$\mathrm{sim}(x,\ y) = \sigma(\Phi x \cdot y + b)$$

ただし，$\sigma(\cdot)$ はシグモイド関数，b はバイアス，y' は x の上位語でない単語である．この手法では，教師データ (上位語・下位語ペア) を関係性ごとにクラスタリングし，クラスタごとに 1 つの線形写像行列を用意することで下位語から上位語への多様な関係を捉えている．さらに，クラスタリングと線形写像行列を同時に学習する独自の最適化アルゴリズムによって高精度に上位語を推定している．しかしながら，これらのモデルは下位語から上位語への線形関係をモデル化しており，非線形性を考慮していない．さらに上位語を推定することに焦点を絞っており，上位語から下位語への関係を考慮していない．そのため，「上位語の下位語には元の単語が含まれる」などの双方向性に関する制約も考慮していない．

2.3 正準相関分析

正準相関分析 (Canonical Correlation Analysis; CCA) [9] は 2 つの実数値ベクトルの入力に対して，それぞれを線形変換した後の値の相関が最も大きくなるような線形変換を見つける手法である．

CCA は何らかの潜在的相関持つと考えられる 2 種類のデータ行列 X_1，X_2 を入力として，それぞれのデータに対する線形変換行列 A_1 および A_2 を学習する．このとき X_1, X_2 それぞれの共分散行列を Σ_{11}, Σ_{22}，相互共分散行列を Σ_{12} とすると，CCA は以下のように線形変換後の相関を最大化する．

$$\begin{aligned} &\text{maximize:} \quad \mathrm{tr}\left({A_1}^{\mathrm{T}}\Sigma_{12}A_2\right) , \\ &\text{subject to:} \quad {A_1}^{\mathrm{T}}\Sigma_{11}A_1 = {A_2}^{\mathrm{T}}\Sigma_{22}A_2 = I \end{aligned} \tag{2}$$

ここで，$\mathrm{tr}(A)$ は行列 A の対角成分の和である．

この最適化問題は解析的に解くことが出来る．行列 $T \equiv \Sigma_{11}^{-1/2}\Sigma_{12}\Sigma_{22}^{-1/2}$ を定義し，U_k と V_k をそれぞれ T の左特異ベクトル，右特異ベクトルとする．このとき最適なパラメータ $(A_1^*,\ A_2^*)$ は，

$$(A_1^*,\ A_2^*) = \left(\Sigma_{11}^{-1/2}U_k,\ \Sigma_{22}^{-1/2}V_k\right) \tag{3}$$

となる．

2.4 深層学習に基づく正準相関分析

深層学習に基づく正準相関分析 (Deep Canonical Correlation Analysis; DCCA) [4] は共通要因を持つと考えられる 2 種類 (view1 と view2) のデータを非線形変換を含む多層パーセプトロン (Multi Layer Perceptron; MLP) で変換し，変換後のデータ内に含まれるペアの相関係数の和ができるだけ大きくなるようにパラメータを最適化する手法である．図 1 にその概要を示す．

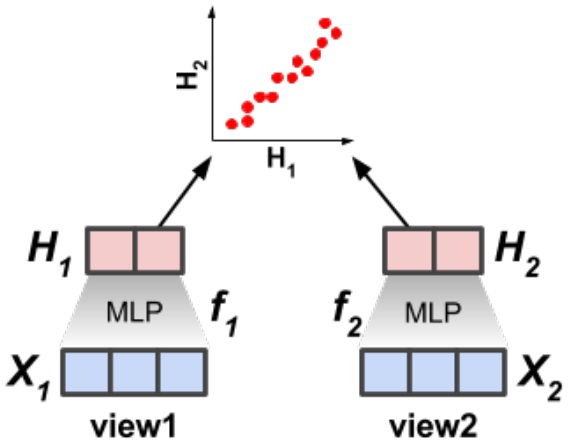

図 1: DCCA の概要

2 種類のデータ行列 X_1，X_2 を，それぞれを対応する MLP (f_1, f_2) で変換した行列 H_1，H_2 は，

$$H_1 = f_1(X_1;\ \theta_2)\ , \quad H_2 = f_2(X_2;\ \theta_2) \tag{4}$$

となる．ただし，θ_1，θ_2 はそれぞれの MLP の学習パラメータである．このとき H_1 と H_2 それぞれの共分散行列を Σ_{11}, Σ_{22}，相互共分散行列を Σ_{12} とすると，H_1 と H_2 の相関係数の和は，

$$\mathrm{tr}\left(T^{\mathrm{T}}T\right)^{-1/2} \tag{5}$$

となる．(5) 式を最大化する学習パラメータ θ_1，θ_2 は解析的には求まらないため，誤差逆伝搬により (5) 式の値が出来るだけ大きくなるように更新される．

CCA では入力の線形関係しか捉えられないのに対して，DCCA は MLP により非線形関係を含む複雑な関係を捉えることが出来ると考えられており，様々なタスクで CCA より良い性能を示している．

3 提案手法

本節では DCCA を利用することで非線形性および双方向性を考慮した相関分析に基づく上位語・下位語関係推定手法を提案する．

分類器によるモデルは潜在的に lexical memorization という問題点を回避しながら，特徴ベクトルを人手で設計する必要があるため，最適な設計を見つけることは容易ではない．また，線形写像に基づく上位語

推定モデルは非線形性と上位語・下位語間の双方向性を考慮できていない．

以降，提案手法の詳細について説明する．いま，上位語・下位語の分散表現のペア $(X,\ Y)$ を考える．$X \in \mathbb{R}^{N\times d}$ は各行に下位語の分散表現を並べた行列であり，N はペアの総数，d は単語の分散表現の次元である．また，$Y \in \mathbb{R}^{N\times d}$ は各行に上位語の分散表現を並べた行列である．

- **学習ステップ**
  DCCA によって変換された $X,\ Y$ を $X',\ Y'$ とすると，

$$\begin{aligned} X' &= f_1(X;\ \theta_1)\ , \\ Y' &= f_2(Y;\ \theta_2)\ , \end{aligned} \tag{6}$$

  となる．ただし，$f_1,\ f_2$ は $X,\ Y$ に対する MLP を関数で表したものであり，$\theta_1,\ \theta_2$ はそれぞれの関数のパラメータである．$\theta_1,\ \theta_2$ は X' と Y' の相関を大きくするように 2.4 節で述べた方法により最適化される．

  その後，DCCA の出力に対して CCA を適用する[1]．つまり，学習を終えた DCCA によって変換された $X,\ Y$ に対して相関が最大になるような線形写像 $A_1,\ A_2$ を 2.3 節で述べた方法により求める．

- **推定ステップ**
  2 つの単語の分散表現を $w_1,\ w_2$ とするとき，

$$\begin{aligned} &\mathrm{sim}_{DCCA}(w_1,\ w_2) = \\ &\quad \mathrm{cosine}\left(A_1\ f_1\left(w_1\right),\ A_2\ f_2\left(w_2\right)\right)\ , \end{aligned} \tag{7}$$

  を定義する．この値がしきい値 λ よりも大きければ，w_2 は w_1 の上位語であると推定する．

線形写像に基づく手法では，単語の分散表現を同ベクトル空間上に存在する上位語の分散表現に写像するものと考えることができる．一方，提案手法は上位語および下位語の分散表現を DCCA と CCA の変換によって元のベクトル空間とは別の新たなベクトル空間に写像する．写像では，MLP による非線形変換を行うため，提案手法は非線形性を考慮することができる．また，語彙内の全ての分散表現を新たなベクトル空間に写像すれば，そのベクトル空間において任意の単語の近傍を探索するだけでその上位語および下位語を同時に推定することが出来るため，双方向性を考慮したモデルであると考えられる．

[1] これは [4] を始めとして，DCCA に基づく複数の研究で行われている．

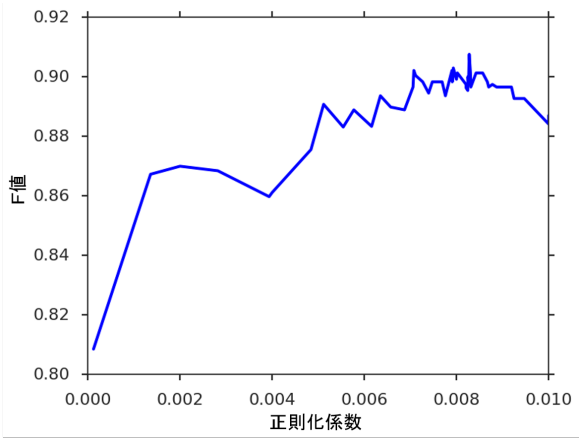

図 2: 正則化係数 (横軸) と開発データにおける F 値 (縦軸) の関係．

4 実験

上位・下位語関係の双方向性および非線形性を考慮することの効果を検証するために以下の実験を行った．

4.1 実験設定

単語の分散表現は Web に公開されている Skip-gram with Negative Sampling [1] で学習されたもの[2]を用いた．学習コーパスは GoogleNews dataset (10 兆単語) で，分散表現の次元は 300 である．計算コストが大きくなるのを防ぐために，分散表現学習の際のコーパスにおける出現頻度上位 10 万単語の分散表現を学習及び実験で用いた．

上位語・下位語ペアのデータセットは LEDS [7] を用いた．このデータセットには正例と負例がそれぞれ 1,385 個含まれており，Levy ら [10] らの実験と同様に学習・開発・評価データに分割した．ただし，この中から上位語および下位語がどちらも分散表現の語彙に含まれるものを用いた．

最適化アルゴリズムは Adam を用い，MLP の層数および各層の次元，L2 正則化係数はベイズ最適化を用いて開発データで調整を行った．

本実験では，提案手法，CCA (提案手法の DCCA を CCA にしたもの)，線形写像に基づく手法 [2]，分類器に基づく手法 [10] の 4 つのモデルを，与えられた 2 単語が上位語・下位語の関係にあるか否かを二値で分類した際の F 値で評価・比較した．

[2] https://drive.google.com/file/d/0B7XkCwpI5KDYNlNUTTlSS21pQmM

4.2 実験結果および考察

評価データにおける各手法の F 値を表 1 に示す．提案手法が最も良い F 値を達成している．これより，写像行列に基づく手法や，CCA に基づく手法は非線形性を考慮していないため上位・下位語関係の双方向性を捉えることが出来ないが，非線形性を考慮した提案手法は双方向性を捉えることが出来ていると考えられる．

図 2 に提案手法における正則化係数と開発データでの F 値の関係を表すグラフを示す．図より正則化を行うことで提案手法の汎化性能が向上することが分かる．

また，提案手法は (7) 式の値が大きくなるような単語を探索することで，与えられた単語の上位語や下位語を生成する事ができる．表 2 に “beer” の上位語として生成された単語を (7) 式の値が大きい順に示す．また，主観評価によると提案手法が CCA と比較して上手く上位語を生成できていることが分かる．

5 おわりに

上位語・下位語関係の双方向性と非線形性を考慮するために，DCCA に基づく手法を提案・評価し，それらを考慮していない手法との比較を行った．実験の結果，提案手法が他のベースラインと比較して大幅に F 値を向上できることがわかった．また，提案手法は上位語にも応用することができ，主観評価によれば線形モデルよりも上手く上位語を生成できることがわかった．今後の展望として，上位語・下位語関係を明示的に分散表現学習に取り入れる新たな手法を考案することが考えられる．

表 1: 評価データにおける二値分類の F 値．

手法	F 値
線形写像に基づく手法 [2]	0.766
CCA に基づく手法	0.747
分類器に基づく手法 [10]	0.802
提案手法	0.824

表 2: 提案手法および CCA により生成された “beer” の上位語 (太字は主観評価による正解)．

順位	提案手法	CCA
1	**liquid**	**liquid**
2	**fluid**	**fluid**
3	**beverage**	cristal
4	**liquids**	**bevarage**
5	soft_drink	gilmer
6	**fruids**	bottled
7	**alcohol**	bud
8	diet_coke	pills
9	beer	buttermilk
10	milkshake	**alcohol**

参考文献

[1] Tomas Mikolov et al. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119, 2013.

[2] Josuke Yamane et al. Distributional hypernym generation by jointly learning clusters and projections. In *COLING*, pages 1871–1879, 2016.

[3] Ruiji Fu et al. Learning semantic hierarchies via word embeddings. In *ACL*, pages 1199–1209, 2014.

[4] Galen Andrew et al. Deep canonical correlation analysis. In *ICML*, pages 1247–1255, 2013.

[5] Manaal Faruqui et al. Improving vector space word representations using multilingual correlation. In *EACL*, pages 462–471, 2014.

[6] Ang Lu et al. Deep multilingual correlation for improved word embeddings. In *NAACL*, pages 250–256, 2015.

[7] Marco Baroni et al. Entailment above the word level in distributional semantics. In *EACL*, pages 23–32, 2012.

[8] Stephen Rolleret al. Relations such as hypernymy: Identifying and exploiting hearst patterns in distributional vectors for lexical entailment. In *EMNLP*, pages 2163–2172, 2016.

[9] Harold Hotelling. Relation between two sets of variables. *Biometrika*, 28(3):321–377, 1954.

[10] Omer Levy et al. Do supervised distributional methods really learn lexical inference relations? In *NAACL-HLT*, pages 970–976, 2015.