分散表現を用いた語の上位下位関係の学習 ーLexical Memorizationの緩和ー

鷲尾 光樹* 加藤 恒昭†
東京大学 大学院総合文化研究科
kokiwashio@g.ecc.u-tokyo.ac.jp* ; kato@boz.c.u-tokyo.ac.jp†

1 はじめに

分散表現を用いた計算機による語の意味関係の学習・識別は、含意関係認識やシソーラスの自動的構築など、高度な意味処理を必要とするタスクにおいて重要である。共起情報に基づいてコーパスから自動的に獲得される語の分散表現を用いることで、人手で作られたリソースにない語に関しても、意味関係が識別できる。従来より、この技術を用いて二語の類義性を測ることは行われているが、近年では類義関係以外にも、様々な意味関係に関する研究が行われている。本稿では、分散表現を用いた上位下位関係の教師あり学習を取り扱う。

上位下位関係とは、「車」-「乗り物」や、「りんご」-「果物」といった、“A is a B” と言えるような二語の意味関係である。これを計算機に識別させる方法としては、二語の分散表現を特徴とし、訓練データを用いて学習を行う教師あり学習が主流である。しかし、この方法は、一見高い性能を達成しているように見えて、実際には汎化性能に致命的な欠陥があることが報告されている [3]。この現象は、Lexical Memorization と呼ばれている。本稿ではこの現象に関して問題点を整理し、未解明の原因について明らかにするとともに、現状の教師あり学習を改善する手法を提案する。さらに、実験により提案手法を用いることで、汎化性能が向上し、実際に問題の現象が緩和されていることを示す。

2 背景

2.1 上位下位関係の学習

語の分散表現を用いた上位下位関係の学習には二つのアプローチがある。ひとつは、訓練データを用いずに、二語のベクトルから上位下位関係性を識別するような指標を用いる教師なし学習である。もうひとつは、二語のベクトルを特徴とし、シソーラスなどから抽出したデータを用いて学習を行う教師あり学習である。

教師なし学習では、内省的な直観に基づいた指標を用いて、二語が上位下位関係にあるか否かの分類を行う。これらの指標は、分布包含仮説に基づくものが主流である。分布包含仮説とは、下位語の出現文脈は上位語の出現文脈に包含されているはずだという直観である。これに基づいて、単語ベクトルの各次元の値を比較することで、二語の上位下位関係性を識別できる [2]。他にも、共起頻度ベクトルを確率分布として捉え、分布の形状をエントロピーを用いて比較することで、二語の意味の広さを測り、上位下位関係性を識別するような指標がある [5][7]。このアプローチにおいては、ベクトルの各次元の意味が明確である必要があるので、古典的な共起頻度ベースのベクトルを分散表現として用いる。

一方、教師あり学習では、二語の分散表現から特徴ベクトルを作り、単語のペアに意味関係が付与された訓練データを用いて、分類器を獲得する。特徴ベクトルの作り方としては、二つの分散表現の差を用いる **DIFF** と、連結を用いる **CONCAT** がある。分類モデルは、線形分類器がよく用いられる。非線形分類器は、過学習の起こりやすさ等から、データに限りのある上位下位関係の学習においては線形分類器に劣るケースが多い [4]。

教師あり学習は、訓練データを用いる分、基本的に教師なし学習よりも性能がよいが、Lexical Memorization という現象が報告されており、汎化性能に欠陥があることがわかっている。

2.2 Lexical Memorization

Lexical Memorization（以下 LM）とは、Levy らの研究 [3] で明らかになった、上位下位関係の教師あり学習の問題点である。これは、DIFF と CONCAT に共通して観測されている現象である。具体的には、次の二つの分類器の振る舞いを指している。

1. 未知語で構成されたペアに対する汎化性能の低下

2. switched pair（ちぐはぐな上位下位ペア）の正例への分類

第一の問題は、訓練データとテストデータの語彙の重なりをなくした場合（この処理を Lexical splitting と呼ぶ)、通常のランダムな交差検定に比べて性能が大幅に下がってしまったことから確かめられた。さらに、上位語位置のベクトルのみで学習した場合と、二語のベクトルを用いた場合とで、性能の差が僅かであったことから、分類器はほとんど上位語位置のベクトルの情報しか見ていないことがわかった。以上から、分類器は訓練データ内の上位語に過剰適合していると解釈できる。

第二の問題は、DIFF や CONCAT などを含めた様々な条件で学習させた分類器に、「りんご」-「乗り物」のような switched pair を分類させると、それらを正例に分類してしまう割合が recall とほぼ等しく、正例と switched pair を区別できていないことから確かめられた。Levy らは、その原因を、DIFF や CONCAT を用いた線形分類器が、二語の関係性を見ていないことに求めている。いま、DIFF と CONCAT、それぞれの場合のパラメータ θ と二語のベクトル $\vec{w_1}$、$\vec{w_2}$ の積を見ると、以下のようになる。

$$\begin{aligned} DIFF(w_1, w_2; \theta) &= \theta(\vec{w_2} - \vec{w_1}) \\ &= \theta \cdot \vec{w_2} - \theta \cdot \vec{w_1} \end{aligned} \quad (1)$$

$$\begin{aligned} CONCAT(w_1, w_2; \theta) &= \theta(\vec{w_1} \oplus \vec{w_2}) \\ &= \theta_1 \cdot \vec{w_1} + \theta_2 \cdot \vec{w_2} \end{aligned} \quad (2)$$

ただし、CONCAT の場合、$\theta = \theta_1 \oplus \theta_2$ である。

上の二つの式を見ると、$\vec{w_1}$ と $\vec{w_2}$ の関係性を捉えるような $\vec{w_1} \cdot \vec{w_2}$ 項は存在しない。これら二つのモデルは二語の関係性自体は見ていないため、switched pair を正例として分類してしまうと Levy らは分析し、線形分類器を用いた教師あり学習では、二語の関係性を学習できず、「典型的な上位語」を覚えているに過ぎないと結論づけている。

この現象は、人手で作られたリソースにない事例に関しても、意味関係が識別できるという、分散表現を用いたアプローチのメリットが損なわれてしまうという点で、致命的である。

3 上位語への過剰適合

本稿ではこの LM の原因を考察し、その対処法を提案する。前述の通り、LM は、「分類器が二語の関係性を学習せず、訓練データ内の上位語に過剰適合し、汎化性能が低下してしまう」現象である。「二語の関係性を学習しないこと」と、「訓練データ内の上位語に過剰適合してしまうこと」は、表裏一体の現象のように思われるが、本稿ではこれらの二つの問題が、それぞれ別の原因を持つことに着眼する。前者の原因に関しては、2.2 節で、述べた Levy らの分析ですでに説明されている。しかし、後者の問題は、それだけでは説明できない。

式 (1)、式 (2) を見ると、これらのモデルは二語の関係性は捉えられないものの、DIFF の場合は上位語ベクトルと下位語ベクトルの典型的な位置関係、CONCAT の場合は典型的な下位語と典型的な上位語のベクトルが捉えられるはずである。もし理想的な学習が行われ、一般性を持つ典型性が学習できたならば、switched pair には対応できないものの、未知語で構成されたペアに対する汎化性能の低下は起こらないはずである。以上から、上位語への過剰適合の原因は、学習モデルではなくデータに求める必要があると考えられる。

3.1 原因分析

上位語への過剰適合に関わりそうなデータの性質として、正例の上位語の出現回数の偏りが挙げられる。上位下位関係データの元となるシソーラスは、基本的にツリー構造を持ち、広い意味を持つ語ほど多くの下位語を持つようになっている。結果として、シソーラスからナイーブに上位下位関係ペアを抽出した場合、下位語を多く持つ語ほど、データセットの上位語位置に出現し、上位語位置にくる単語の出現回数は大きく偏ることになる。この性質が、DIFF と CONCAT それぞれについてどのような悪影響を及ぼすかについて、以下の仮説が立てられる。

DIFF は二語のベクトルの差を特徴ベクトルとするモデルである。いま、式 (1) の w_1 を下位語、w_2 を上位語とする。訓練データの正例の上位語には、特定の語が何度も出現し、重複が多く種類が少ないため、一定の傾向があると思われる。一方、下位語位置にある語は重複が少なく、上位語ほど傾向がないはずである。ゆえに、DIFF では正例に対して、$-\theta \cdot \vec{w_1}$ を考慮せず、何度も出現する w_2 に対して、$\theta \cdot \vec{w_2}$ のみを大きくするように学習してしまうと考えられる。これによって、多くの上位下位関係ペアに共通する典型的な特徴以上に、訓練データに何度も出現する特定の上位語のドメインなどの情報に過剰適合してしまうのだと思われる。

次に CONCAT の場合について述べる。CONCAT は、二語のベクトルの連結を特徴ベクトルとするモデルである。正例の上位語位置にある語に重複が多い場合、ある一定の傾向を持ったベクトルが何度も上位語位置の特徴（$\vec{w_2}$）に正例として入力されることになる。この結果、$\vec{w_1}$ が無視され、θ_2 のみが大きくなっていき、上位語と

表 1: WeedsBLESS: 正例の語の出現回数の統計量

	平均	分散（標準偏差）	中央値	最頻値
上位語	7.7	139.7(11.8)	4	1
下位語	4.3	3.5(1.9)	4	4

下位語の典型性をそれぞれ捉える以上に、特定の上位語の情報に対し分類器が重み付けしてしまい、過剰適合が引き起こされるのだと考えられる。

3.2 検証

3.1 節で述べた仮説の妥当性を検証するために、英語名詞を対象に、上位語位置にある語の出現回数の偏りが汎化性能に対して及ぼす悪影響についての実験を行った。

語のペアに意味関係が付与された訓練用データセットとして、WeedsWN と WeedsBLESS[8] を用いた。WeedsWN は WordNet から作られたデータセットであるが、特徴的な点は、データセット内の各語が、上位語位置と下位語位置に 1 回ずつしか出現しないように、制約が課されているところである。一方、WeedsBLESS は、BLESS というデータセットから作られているが、そのような制約がない。表 1 は、WeedsBLESS の正例の、上位語位置と下位語位置それぞれにおける、語の出現回数の統計量である。これを見ると、上位語の出現回数の分布には大きな偏りがあり、下位語においては偏りが少ないことがわかる。WeedsWN と WeedsBLESS でそれぞれ学習した分類器の性能を比較することで、上位語位置にある語の出現回数の偏りが汎化性能に対して及ぼす悪影響を調べることができる。

分散表現には、Omer Levy が公開している、近傍共起前後 2 語を文脈とした分散表現を用いた[1]。これは英語版 Wikipedia から出現頻度 100 以下の語を無視して、Skipgram モデルを用いて獲得された分散表現である。

分類器には L2 正則化を行うロジスティック回帰を用いた。

テストデータには、Hyperlex[6] から、Lexical splitting のために、WeedsBLESS と WeedsWN に出現した語で構成されていないペアを抽出して用いた。Hyperlex は Wordnet などのリソースから様々な意味関係を持つペアを集めたものである。結果として、141 の正例と 239 の負例での検証となった。

まず、CONCAT を用いた場合の検証結果を表 2 に示す。WeedsBLESS で学習した場合、recall が顕著に低く、語の出現回数に制約がある WeedsWN と比べて、汎化性能に大きく差があることがわかる。これは、訓練データへの過剰適合と解釈できる。

[1]https://levyomer.wordpress.com/2014/04/25/dependency-based-word-embeddings/

表 2: 各データセットで訓練した場合の性能比較

	precision	recall	F1
$WeedsWN$	0.508	**0.702**	0.589
$WeedsBLESS$	0.761	**0.113**	0.198

表 3: 分類器のパラメータの二乗の平均

	下位語位置	上位語位置
$WeedsWN$	0.349(0.591)	0.550(0.742)
$WeedsBLESS$	0.097(0.312)	0.750(0.866)

さらに、WeedsWN で学習した分類器と、WeedsBLESS で学習した分類器の上位語位置と下位語位置のパラメータの二乗の平均を、表 3 に示す。括弧内は値の平方根である。これを見ると WeedsBLESS で学習した分類器は、上位語の情報に偏った重み付けをしていることがわかる。以上の結果から CONCAT における訓練データ内の上位語への過剰適合の傾向が見られる。

DIFF の場合でも、表 2 のような分類器の性能差は同じ傾向であった。DIFF においても WeedsBLESS で学習した際の上位語への過剰適合の傾向を確かめるために、訓練データ内の正例の上位語の出現回数と、正例の特徴ベクトルとパラメータベクトルの内積の相関を調べた。本来ならば、上位語の出現回数と学習された上位下位関係性は無相関であるのが好ましいが、相関係数が **0.903** であり、非常に強い相関があることがわかった。

以上の検証から、LM の上位語への過剰適合という問題の原因が、訓練データ内の上位語の出現回数の偏りにあることがわかった。

4 提案手法と実験

4.1 提案手法

2.2 節の Levy らの分析と、3 節の我々の分析を踏まえ、LM の二つの問題に対して、それぞれに対応する二つの手法を提案する。

第一の手法は、訓練データ内の上位語への過剰適合と未知語ペアへの汎化性能の低下を防ぐためのものであるが、3 節で考察した通り、この原因は訓練データの偏りにあるので、訓練データ内の各位置における出現回数に制約を課すことで対応する。これを Lexical Occurence Constraing(以下 LOC) と名付ける。

第二の手法は、二語の関係性を捉えていないことに対処するためのものであるが、Levy の指摘通り、そもそも DIFF や CONCAT では、本質的に対応が不可能であるので、二語の関係性を捉える指標として研究されてきた、教師なし学習の指標を特徴ベクトルに追加する。これを Feature of Unsupervised Measures(以下 FUM) と

表 4: 性能比較

	precision	recall	F1	match error
DIFF	0.723	0.752	0.735	0.393
LOC 適用	0.690	0.820	0.748	0.467
FUM 適用	**0.742**	0.783	0.761	**0.236**
ALL	0.716	**0.832**	**0.768**	0.346

名付ける。以下では、提案手法を評価するための実験について述べる。

4.2 実験

分散表現や分類器には 3.2 節と同じものを用いた。FUM で教師なし学習の指標を用いるためには共起頻度ベースのベクトルが必要なので、Levy らの分散表現と同じく、英語版 Wikipedia から近傍共起前後 2 語を文脈とし、出現回数 100 以下の語を無視して獲得した PPMI ベクトルを用いた[2]。

データセットには、Hyperlex と、switched pair を負例として多く含む LEDS[1] を、重複するペアを取り除いた上で、合わせて用いた。このデータセットを、文献 [4] と同じ方法で、Lexical splitting しながら分割し、30fold の交差検定用データを作った。そして、各 fold において、分散表現が獲得できたペアのみを用いて評価を行った。分散表現が獲得できたペアは 4811 事例であった[3]。

LOC を適用する場合は、訓練データにおいて、まず正例の各語の上位語位置、下位語位置の出現回数が 1 回ずつになるように制約をかけ、その後、各位置にまだ出現していないペアで構成される負例を訓練データに追加していった。FUM を適用する際は、特徴ベクトルに、教師なし学習指標として用いられてきた、コサイン類似度、invCL[2]、二語の分布のエントロピーの差 [5][7] を特徴に追加した[4]。これらの指標はそれぞれ、二語の類似度、分布の包含性、分布の形状の違いを見ている。

結果を表 4 に示す。なお、CONCAT を用いた場合の結果も似た傾向にあったため、DIFF の結果のみを報告している。ベースラインは DIFF のみを用いた場合である。なお、match error とは、switched pair を誤って正例に分類してしまった割合である。まず、提案手法を適用した場合、ベースラインと比べて F1 が向上している。いずれの差も統計的に有意であった[5]。LOC を適用することで、recall が大幅に上がり、汎化性能が向上していることがわかる。また、FUM を適用することで、precision、recall が向上し、match error が最も低くなっており、関係性の学習が促進されている。これらの結果から、提案手法が LM の緩和に有効であることが示された。さらに FUM と LOC を両方適用した ALL では、recall、F1 がもっとも良かった。しかし、ALL は FUM のみ適用した場合と比べて match error が高く、F1 の差も統計的に有意ではなかった[6]。これは、LOC によって、訓練データが大幅に減ってしまっており、関係性の学習が FUM のみ適用する場合に比べて促進されなかったためであると思われる。

[2]獲得には Omer Levy が公開している hyperwords を用いた。https://bitbucket.org/omerlevy/hyperwords

[3]総数は 4821 事例だが、うち 10 事例については、利用した分散表現にその中の語が含まれていなかったので取り除いた。

[4]コサイン類似度の計算には、Skipgram ベクトルを用いた。

[5]ウィルコクソンの符号順位検定、DIFF と DIFF+LOC は $p < 0.05$、それ以外は $p < 0.01$。

5 結論

本研究では、語の上位下位関係の教師あり学習の問題である、LM について、問題点を整理し、その原因を明らかにした。さらに、各問題を解決するための手法を提案し、その有効性を実験により示した。

今後の課題について以下で述べる。LOC は分類器の汎化性能を向上させる、非常に単純で効果的な手法であるが、訓練データを大幅に減らしてしまうという欠点がある。結果として、FUM と合わせて用いた場合、関係性の学習を促進させる FUM の利点を活かしきれない。これを解決するために、今後は訓練データを減らさずに、データ内の各語の影響力を調節する手法について検討したい。

参考文献

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

[2] Alessandro Lenci and Giulia Benotto. Identifying hypernyms in distributional semantic spaces. In **SEM*, pp. 75–79, 2012.

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

[4] Stephen Roller and Katrin Erk. Relations such as hypernymy: Identifying and exploiting hearst patterns in distributional vectors for lexical entailment. In *EMNLP*, pp. 2163–2172, 2016.

[5] Enrico Santus et al. Nine features in a random forest to learn taxonomical semantic relations. In *LREC*, pp. 4557–4564, 2016.

[6] Ivan Vulić et al. Hyperlex: A large-scale evaluation of graded lexical entailment. *arXiv preprint arXiv:1608.02117*, 2016.

[7] 鷲尾光樹. 語の分散表現と上位下位関係—研究動向と今後への試案—. 第 14 回インタラクティブ情報アクセスと可視化マイニング研究会, pp. 14–21, 2016.

[8] Julie Weeds et al. Learning to distinguish hypernyms and co-hyponyms. In *COLING*, pp. 2249–2259, 2014.

[6]CONCAT の場合、CONCAT+FUM の F1 は 0.771 で、ALL は 0.769 であり、FUM のみ適用した場合のほうが僅かに良かったが、この差も統計的に有意ではなかった。