言語モデルを用いた上位語・下位語関係の推定

橋本 隼人[1]　森 信介[2]
[1] 京都大学 情報学研究科　[2] 京都大学 学術情報メディアセンター

1　はじめに

上位語・下位語関係の推定は、人工知能のための世界知識の構築のために重要である。いくつかのベンチマーク [2, 18, 3, 23, 22] が提案されている。上位語・下位語関係の推定タスクは、v が w の上位語である（$v \prec w$）ような単語ペアを、そうでないペアから区別する 2 値分類問題とみなすことができる。しかし、このタスクは、上位語として現れやすい単語を覚えるだけの分類器を高く評価すると批判されている [19]。また、このようなタスクでは、評価データセットによって最良のモデルは一貫していない傾向にある。そのような傾向は、例えば [6] における表で確認することができる。これは、データセットが含む誤り例の作り方が大きく異なっているためと考えられる。結果として、ランダムな単語のペアであったり、全体語・部分語のペアが選ばれていたり、上位語・下位語を反転させて作られていたりする。

最近では、上位語発見（Hypernym Discovery）という新しい評価のフレームワークが提案されている [8]。上位語発見は、与えられた下位語に対して、コーパスに現れる（ほとんど）すべての語句の中から適当な上位語を見つけ出すタスクである。このような評価は、実際の応用タスクに近いものであることから、本研究は SemEval 2018 shared task [5] により実験する。

我々は、次節で説明する研究背景から、単純かつ直截的なアプローチによって、単語の意味を 1 つのベクトルで表す方法よりも高い性能が期待できる方法に思い至った。それは、LSTM [11] LM によって v の代わりに w という語句がどの程度当てはまりやすいかを計算し、その結果を $v \prec w$ であるかどうかの指標に用いる方法である。

我々はこの手法を実装し、SemEval 2018 shared task により評価した。以下の節では、研究背景、手法の詳細、実験結果と考察を述べる。

2　背景

分布包含仮説（distributional inclusion hypothesis; DIH) [9] は、上位語・下位語関係の教師なし学習を可能にする理論的基盤である。これは分布仮説 [10] を拡張するものである。この仮説に基づいて、語句の共起行列や PPMI 行列 [4] から上位語・下位語関係らしさを計算するいくつもの指標が提案されてきた [24, 7, 13, 14]。中でも、分布情報量仮説 [17] は、DIH から発想された別の理論的基盤である。この仮説は、「ある下位語を含む文はその単語を上位語に置き換えられた文よりも（情報理論によって定義される）情報量が多い」としている。また、予測モデルから語句の意味論へのアプローチとして、最近いくつかの研究がある。 Baroni et al. [1] は、Skip-Gram の予測タスクにより学習された語句の埋め込みベクトル表現は、計数ベースの手法よりも語句の関連性やアナロジーの推定でよりよいことを示した。上位語・下位語関係では、Chang et al. [6] が本関係をとらえるために設計された Skip-Gram の予測モデルを用いた手法 (distributional inclusion vector embedding; DIVE) を提案し、先述の 2 値分類問題で評価し良い結果を示した。これは、上位語発見タスクにおいて、合理的なベースラインであると考えられる。

3　言語モデルに基づく上位語の程度

3.1　分布包含仮説と言語モデル

DIH とは、「w が v の上位語ならば ($v \prec w$)、w のすべての統語論的特徴は v でも成り立ち、かつ、その逆も真[*1]」という主張である。先行研究では、統語論的特徴として、ある Bag-of-Words に単語がどれだけ当てはまりやすいか、を挙げ、これによって DIH がどれほど成り立っているのかの指標としている。

我々は、 単語 v を空白にしたテキスト（これを**具体的な文脈**と呼ぶことにする）にどれほど当てはまりやすいか、を統語論的特徴として使うことを提案する。例えば、“Antiviral drugs are ____ in treating influenza.” は v = “effective” の **具体的な文脈**である。語彙 V に含まれるすべての単語 w について、v の**具体的な文脈**への平均的な当てはまりやすさを計算し、DIH が成り立っていることの指標として用いる。この指標を w の v への語句代替性と呼ぶことにする (3.4 節で、語句代替性の具体例として **LM-Measure** を定義する)。**具体的な文脈**は、まったく同一のものはほぼ存在しないので、計数によって語句代替性を計算することはできない。そうではなく、言語モデル (以下では LM と表記) を用いて語句が空白を埋めることのできる確率を計算することを提案する。

3.2　Contextual Language Models

普通の LM は、先行する単語列のみを手がかりに単語を予測する。このようなモデルは、単語の品詞を間違いやすい傾向にある。3.1 節の例でいえば、LSTM LM は “not” “generally” “the” と言った単語を予測する。したがって、我々は前後両方の単語を手がかりにする contextual language model (cLM) を用いることにした。次節で述べられる方法で学習した結果、“effective”

[*1] DIH の原著論文 [9] においては、多義性解消後の単語に対して主張しているが、本論文では簡単のために単語の多義性を無視する。

“useful” “recommended” といった正しい品詞の単語を予測することを確認した。

3.3 Contextual Bidirectional LSTM の学習

Contextual BLSTM [16] (c BLSTM) は、前向き後ろ向きの 2 つの LSTM を組み合わせた cLM である。この手法では、確率分布のロジットベクトルは $\boldsymbol{h} = \boldsymbol{h}_{f,k-1} + \boldsymbol{h}_{b,k+1}$ と計算した（それぞれ、前向きの $(k-1)$ 番目と $(k+1)$ 番目の出力を表す）。[12] に従い入力と出力の埋め込みパラメータを共有し、クロスエントロピー最小化により学習した。

3.4 cBLSTM による上位語の程度

分布情報量仮説からの類推により、w の v への語句代替性 **LM-Measure** $(v \prec w)$ を v のすべての**具体的な文脈** $C[v]$ について平均した対数尤度の差、つまり相対情報量として定義する。

$$\textbf{LM-Measure}\,(v \prec w) = \mathop{\mathbb{E}}_{c \in C[v]} \left[\log P_{\mathbf{LM}}(w|c) - \log P_{\mathbf{LM}}(v|c)\right]$$

我々は、この値を 3.1 節で議論した上位語らしさとして用いた。ところで、単語の確率分布を softmax 関数に渡すロジットベクトル $\boldsymbol{h}$ として推定する LM や cLM では、相対的情報量 $\log P_{\mathbf{LM}}(w|c) - \log P_{\mathbf{LM}}(v|c)$ は以下のように計算できる。

$$\begin{aligned}&\log[\mathrm{softmax}(\boldsymbol{h})]_v - \log[\mathrm{softmax}(\boldsymbol{h})]_w \\ &= \log \frac{\exp(h_v)}{\sum_i \exp(h_i)} - \log \frac{\exp(h_w)}{\sum_i \exp(h_i)} = h_v - h_w\end{aligned}$$

我々は、すべての単語ペアに対する **LM-Measure** をワンパスで計算した。まず、頻度の高い $|V'|$ 単語に含まれるすべての w について、コーパスに v が現れるごとに $h_v - h_w$ を計算した。この値の合計は行列 $W_{\text{Measure}} \in \mathbb{R}^{|V|\times|V'|}$ に記録される。最後に、すべての行を対応する単語頻度で割る。これにより、**LM-Measure** の計算所要時間は cBLSTM の 1 エポック程度に収まる。

4 評価

4.1 学習コーパス

SemEval 2018[*2]の Web サイトより分割済みコーパスを入手した。これは 3 つの一般ドメインの言語別タスクと 2 つの特殊ドメインのためのタスク (2A–医学、 2B–音楽) からなる。我々はコーパスサイズが比較的小さいが、実践的である 2A と 2B のみで実験を行った。うち、最初の 1,000 行を cBLSTM の開発データセットとし、残りを訓練コーパスとした。語彙は、頻度上位 80,000 語と低頻度語を表す<unk>で構成した。

4.2 各モデルの設定

文献 [5] にある手法に加え、教師なしベースライン **WΔS** と **ClogP** を設定した。

WΔS: 我々は上位語の指標 **WΔS** [6] をベースラインとして採用した。**WΔS** は word2vec [15] ベクトルの consine 類似度を、DIVE で定義される一般性シグナル $\Delta S(v \prec w) = \|w\|_1 - \|v\|_1$ で乗算したものである。

我々は、word2vec モデルを Mikolov の実装[*3]によって、DIVE モデルをその著者らの実装[*4]によって学習した。word2vec の次元数 d については最適な値を探索したが、他のハイパーパラメータは固定した (表 2)。

ClogP: 定義により、**LM-Measure** は、高頻度語に高いスコアを与える。そこで、**LM-Measure** が下位語に類似する一般的な単語を選んでいるだけなのかどうかを見るために、以下のベースラインを設定した。

$$\textbf{ClogP}\,[\alpha]\,(v \prec w) = \cos(\boldsymbol{v}, \boldsymbol{w}) + \alpha \cdot \log P(w)$$

ここで、$\boldsymbol{v}$ と $\boldsymbol{w}$ は word2vec による v と w の埋め込み表現であり、α はハイパーパラメータ、$P(w)$ はコーパスにおける w の出現数を全単語数で割った値である。

LM-Measure: cBLSTM は step annealing を用いて 20 エポック学習した。詳細には、コーパスを n 部に分け、それぞれの最初の m 語を 1 バッチとして cBLSTM に与えた。cBLSTM の隠れ状態は次の m を処理するバッチに渡される[*5]。そして、3 節で述べたように **LM-Measure** を $|V'| = 20000$ として計算した。

smoothing: cBLSTM がより多くの種類の単語を予測できるように、[21, 20] に習いラベルの平滑化を行った。我々は、正解・不正解ラベルへの教師信号として 0.9, $0.1/(|V|-1)$ を用いた。

discouting: 一般的に、cLM は高頻度語に高い確率を与える。開発データが使えるならば、w の対数出現頻度に応じて指標を割り引くことで、この影響を打ち消すことができるようにシステムを調整することができる。このバリエーションでは、**LM-Measure** $(v \prec w) - \alpha \cdot \log P(w)$ を上位語の指標として計算した。ここで、α はハイパーパラメータである。

4.3 上位語・下位語対の開発および評価データセット

SemEval 2018 の Web サイトから開発データセットおよび評価データセットを取得した。我々が開発データセットとしているのは、本来教師ありシステムのための学習用データセットである。開発データセットおよび評価データセットは、それぞれ、 1 つの下位語と複数の対応する上位語を 1 対として、500 対が含まれている。そして、それぞれ下位語は、“Concept”(概念) もしくは “Entity”(実体) のいずれかのラベルが付与されている。我々は、データセット 2B から実体を除去し、概念のみで行った (2A はもとより概念のみが含まれる)。これは、実体の上位語である単語の種類が少ないため、分類問題として解くのが妥当であると思われるからである。

[*2] https://competitions.codalab.org/competitions/17119

[*3] https://github.com/tmikolov/word2vec

[*4] https://github.com/iesl/Distributional-Inclusion-Vector-Embedding

[*5] このプロセスでは、後ろ向き LSTM に渡される単語の順序が m 語ごとに乱されるが、この手法により $n \times m$ の単語予測が一回でできる。埋め込み表現及び隠れ状態の次元は 300 とした。

Table 1: タスク 2A と 2B の結果。太文字の値は同じコーパスサイズにおける最良の結果、下線のある値は表中における最良の結果であることを表す。Anu はコーパスと WordNet の両方を用いて学習した結果でることに注意されたい。

	performance - 2A					performance - 2B				
	MAP	MRR	P@1	P@5	P@15	MAP	MRR	P@1	P@5	P@15
利用した学習データ: 100%										
ClogP	13.96	39.14	36.60	12.89	10.31	7.19	20.31	16.20	7.37	5.45
WΔS	3.20	9.21	6.00	3.32	2.45	2.98	8.64	5.03	3.05	2.36
LM-Measure	16.81	42.64	37.40	16.78	12.22	**8.84**	**24.17**	**18.44**	**8.81**	6.58
+ discount	**16.91**	42.77	37.40	**16.98**	**12.28**	**8.84**	**24.17**	**18.44**	**8.81**	6.58
+ smoothing	16.75	**42.82**	**37.60**	16.68	12.14	8.17	21.83	16.76	7.80	**6.61**
+ both	16.75	**42.82**	**37.60**	16.68	12.14	8.17	21.83	16.76	7.80	6.61
教師なし手法による Concept の学習の結果 [5]										
ADAPT	8.13	20.56	-	8.32	-	1.88	5.34	-	1.89	-
Anu	7.05	17.51	-	7.29	-	*10.68*	*27.13*	-	*10.84*	-
(Team 13)	2.55	7.19	-	2.52	-	4.83	14.33	-	4.51	-
balAPInc	0.91	2.10	-	1.08	-	1.44	3.65	-	1.58	-
APSyn	0.65	1.43	-	0.72	-	1.13	2.55	-	1.30	-
SLQS	0.29	0.66	-	0.33	-	0.64	1.25	-	0.65	-

Table 2: ベースラインモデルの学習に用いたハイパーパラメーター

word2vec	
Window Size	8
# of negative samples	25
cbow	1
Downsampling threashold	10^{-4}
Training epochs	15
DIVE	
# of negative samples	30
Inclusion Shift	Yes
Window size	5
PMI filter size	5
Embedding size	100
Training epochs	15

LM-Measure		**ClogP**	
上位語	頻度	上位語	頻度
disease	90	enzyme	2
drug	8	anemia	2
pain	3	pigment	1
disorder	3	neoplasm	1

Table 3: Hypernym discovered unique to each method.

Table 4: “influenza” の上位語の候補

LM-Measure	**ClogP**
Influenza respiratory virus RSV HIV viral cancer pandemic <eos> hepatitis disease H5N1 H1N1 adenovirus lung H3N2 infection infectious human	Influenza H1N1 pandemic RSV ILI pH1N1 viruses outbreak SARI pdm09 ARI outbreaks virus H3N2 H5N1 viral seasonal respiratory flu

4.4 評価の詳細

文献 [5] と同様に、各手法によって対象単語ごとに 15 の上位語候補を得た。それに対し、平均精度 (MAP)、平均逆順位 (MRR)、および k =1 位、5 位、15 位までの精度 (P@k) を正解と比較することで計算した。候補列挙においては、180 のストップワード (基本的に DIVE と同じ) と対象単語自身を除外した。

入力の下位語が句 (単語列) であるときは、最後の単語のスコアを用いて計算した。句を構成する全ての単語の平均ベクトルや平均 **LM-Measure** を用いた実験も行ったが、最後の単語のみを用いたものが性能が良かった。

ハイパーパラメータ α および word2vec の次元数 d は、開発データセットにおいて MAP を最大化するように、それぞれ $\alpha \in \{1/5, 1/10, ..., 1/30, 0\}$ および $d \in \{100, 300, 1000\}$ から選択した。

4.5 結果と議論

表 1 に実験結果を示す。まず、**LM-Measure** が割り引き (discounting) のハイパーパラメータの調節なしに、ベースラインを上回る性能が見られた。これは、上位語であるかを判定するとき、その単語の下位語との類似性と、単語の一般性が手がかりになると考えられるが、分布情報量仮説に基づくデザインを採用したことにより、それらのバランスは手を加えなくともとれている状態であったためであると考えられる。

次に、**LM-Measure** が **ClogP** ベースラインを上回っていることが観察できる。特に、P@5 および P@15 での違いが顕著である。これらの数字は、提案手法が **ClogP** のベースラインよりも、多くの種類の正しい上位語を挙げていることを示唆している。この点を明確にするため、上位語候補 5 語に含まれる正しい上位語をそれぞれのシステムから挙げ、その中もう一方のシステムでは上位語候補 15 語に含まれていないものを抽出した (表 3)。また、具体的に “influenza” の上位語を挙げて調べた (表 4)。それらによると、**ClogP** は共起しやすい関連した単語を上位語の候補として挙げやすいこと、また単語 “disease” を予測することに失敗していることがわかった。これは、筆者が冗語的冗長性を避けた文を書くため、コーパスにおいて疾病の名前と単語「疾病」が共起しにくいためであると考えられる。一方で、**LM-Measure** は “disease” の予測に成功している。

LM-Measure の限界の 1 つは、その手法が下位語の語句代替性のみに基づいていることである。そのためか、**LM-Measure** は “H5N1” の “influenza” 上位語と誤って判定している。

5 おわりに

本論文では、上位語発見の課題に対して、DIH を直接的に実現する手法を提案し、それによって最高精度の結果を得られることを示した。

提案手法の拡張として、句を対象とすることが考えられる。これは、句内の単語境界を特殊な記号で置き換えた上で **LM-Measure** を計算することで簡単に実現できる。他の拡張として、cLM において注意機構を導入することが挙げられる。具体的には Transformer LM [21] を cLM として用いることである。

References

[1] Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 238–247. Association for Computational Linguistics.

[2] Marco Baroni and Alessandro Lenci. 2011. How we blessed distributional semantic evaluation. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics (GEMS '11)*, pages 1–10. Association for Computational Linguistics.

[3] Giulia Benotto. 2015. Distributional models for semantic relations: A study on hyponymy and antonymy. PhD Thesis, Univ. Pisa.

[4] John A Bullinaria and Joseph P Levy. 2007. Extracting semantic representations from word co-occurrence statistics: A computational study. Behav. research methods, 39(3):510–526.

[5] Jose Camacho-Collados, Claudio Delli Bovi, Luis Espinosa Anke, Sergio Oramas, Tommaso Pasini, Enrico Santus, Vered Shwartz, Roberto Navigli, and Horacio Saggion. 2018. Semeval-2018 task 9: Hypernym discovery. In *Proceedings of the 12th International Workshop on Semantic Evaluation*, pages 712–724. Association for Computational Linguistics.

[6] Haw-Shiuan Chang, Ziyun Wang, Luke Vilnis, and Andrew McCallum. 2018. Distributional inclusion vector embedding for unsupervised hypernymy detection. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 485–495. Association for Computational Linguistics.

[7] Daoud Clarke. 2009. Context-theoretic semantics for natural language: An overview. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics (GEMS '09)*, pages 112–119. Association for Computational Linguistics.

[8] Luis Espinosa Anke, Jose Camacho-Collados, Claudio Delli Bovi, and Horacio Saggion. 2016. Supervised distributional hypernym discovery via domain adaptation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 424–435. Association for Computational Linguistics.

[9] Maayan Geffet and Ido Dagan. 2005. The distributional inclusion hypotheses and lexical entailment. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 107–114. Association for Computational Linguistics.

[10] Zellig S. Harris. 1954. Distributional structure. Word, 10(2-3):146–162.

[11] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. Neural Comput., 9(8):1735–1780.

[12] Hakan Inan, Khashayar Khosravi, and Richard Socher. 2017. Tying word vectors and word classifiers: A loss framework for language modeling. In 7th Int. Conf. on Learn. Represent..

[13] Lili Kolterman, Ido Dagan, Idan Szpektor, and Maayan Geffet. 2010. Directional distributional similarity for lexical inference. Nat. Lang. Eng., 16(4):359–389.

[14] Alessandro Lenci and Giulia Benotto. 2012. Identifying hypernyms in distributional semantic spaces. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, SemEval '12, pages 75–79. Association for Computational Linguistics.

[15] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Workshop Proceedings of the International Conference on Learning Representations (ICLR 2013)*.

[16] Amr Mousa and Björn Schuller. 2017. Contextual bidirectional long short-term memory recurrent neural network language models: A generative approach to sentiment analysis. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1023–1032. Association for Computational Linguistics.

[17] Enrico Santus, Alessandro Lenci, Qin Lu, and Sabine Schulte im Walde. 2014. Chasing hypernyms in vector spaces with entropy. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, volume 2: Short Papers*, pages 38–42. Association for Computational Linguistics.

[18] Enrico Santus, Frances Yung, Alessandro Lenci, and Chu-Ren Huang. 2015. Evalution 1.0: an evolving semantic dataset for training and evaluation of distributional semantic models. In *Proceedings of the 4th Workshop on Linked Data in Linguistics: Resources and Applications*, pages 64–69. Association for Computational Linguistics.

[19] Vered Shwartz, Enrico Santus, and Dominik Schlechtweg. 2017. Hypernyms under siege: Linguistically-motivated artillery for hypernymy detection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 65–75. Association for Computational Linguistics.

[20] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[21] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Informa-*

tion Processing Systems 30, pages 5998–6008. Curran Associates, Inc.

[22] Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. 2016. Order-embeddings of images and language. In *Conference Proceedings of the International Conference on Learning Representations (ICLR 2016)*.

[23] Julie Weeds, Daoud Clarke, Jeremy Reffin, David Weir, and Bill Keller. 2014. Learning to distinguish hypernyms and co-hyponyms. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2249–2259. Dublin City University and Association for Computational Linguistics.

[24] Julie Weeds and David Weir. 2003. A general framework for distributional similarity. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing (EMNLP '03)*, pages 81–88. Association for Computational Linguistics.