句の分散表現計算モデルの汎用性の調査

高瀬 翔　岡崎直観　乾 健太郎
東北大学
{takase, okazaki, inui}@ecei.tohoku.ac.jp

1 はじめに

自然言語には単純で小さい要素の組み合わせから複雑で大きい要素が組み立てられていると解釈できる現象が多くある．例えば，文字の組み合わせで単語が作られ，単語の組み合わせで句が作られ，句の組み合わせで文が作られ，文の組み合わせで文章が作られる，といった具合である．ある表現の意味はその構成要素の意味と，それらの合成手続きによって決定される，というフレーゲの構成性原理 [7] も有名である．

Mitchell らは句や文の意味計算の第一歩として，単語のベクトル表現にこの構成性原理を適用し，形容詞–名詞，名詞–名詞，動詞–名詞など，2 単語からなる句のベクトル表現の計算に取り組んだ [9]．さらに近年，ニューラルネットワークを用いて任意長の系列の意味を分散表現（低次元で密なベクトル）に変換するという考えが普及し，文字列から単語の意味を推定する手法や，単語列から文の意味を計算する手法が，機械翻訳や評判分析などで成果を上げている [1]．

しかしながら，現状では，各タスクで高性能を達成することが重視されており，分散表現を合成するモデルが，一般的な意味を計算できるかについての議論はほとんどない．言い換えれば，任意の句の意味計算に適用可能なモデルを作ることが可能か，明らかでない．例えば，Takase らはエンティティ間の関係を表す表現（関係パタン）の分散表現を計算する手法を提案した [5]．関係パタンは “have a great influence on” のように，形容詞–名詞や，“reduce the risk of” のように，動詞–名詞の句を含んでいるが，関係パタンの分散表現を精度良く計算できる手法は，Mitchell ら [9] の構築した，形容詞–名詞など，2 単語からなる句の意味的類似度計算タスクでも高い性能を達成できるのだろうか．

また，Wieting らは言い換え関係認識のため，“in the country of origin” と “in their home countries” や “has no impact” と “is not affected” のような意味的に近い句のペアから，句の分散表現を合成する関数の学習に取り組んだ [4]．彼らのモチベーションに基づけば，学習した関数で分散表現を合成することにより，未知の句のペアでも意味的類似度が測定できる，すなわち，任意の句の意味を表す分散表現が計算できると考えられるが，2 単語からなる句や関係パタンについても，意味的類似度を正しく計算できるのだろうか．

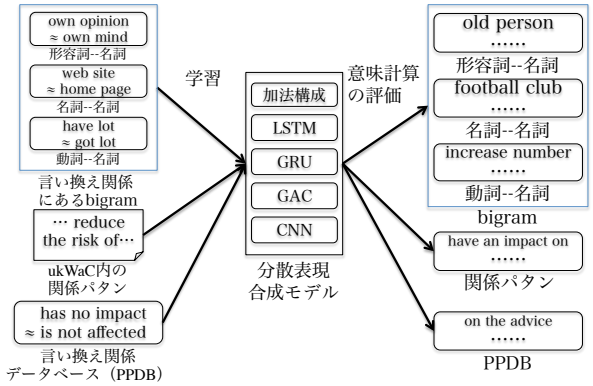

図 1: 本研究の概要図

本研究では，このような，分散表現の合成手法の汎用性を調査する．具体的には，図 1 に示したように，まず，言い換え関係にある Bigram，コーパス中に出現する関係パタン，および，PPDB から抽出した句の言い換え関係を用いて，それぞれの句を対象とした，分散表現の合成モデルを学習する．次に，これらの分散表現合成モデルの，評価データでの性能を検証する．特に，コーパス中に出現する関係パタン，および，句の言い換え関係を用いて学習したモデルについて，学習時とは異なる種類の句に適用した際の性能，すなわち，モデルが汎用的な合成関数を獲得できているかを調査する．

2 評価データ

評価データとして，Mitchell ら [9] のデータセット（形容詞-名詞，名詞-名詞，動詞-名詞の句のペア，それぞれ 108 事例）に意味的類似度の観点から再アノテーションしたもの，“reduce the risk of” と “prevent” のような，関係パタンのペア（5,555 事例）に意味的類似度を付与したもの，“at the recommendation” と “on the advice” のような，言い換え関係データベース（PPDB）[3] から抽出した，3 単語以上 5 単語以下からなる句のペア（1,000 事例）に意味的類似度を付与したものの 3 種類を用いる [4, 5]．

評価としては，まず，各データセットで句のペアに

付与されている類似度と，学習したモデルで計算した句の分散表現間のコサイン類似度とのスピアマンの順位相関係数を計算し，人の判断との相関を測定する．次に，評価データセット間での性能の変化や，学習手法の違いによるモデルの性能を比較する．例えば，全てのデータセットでスピアマンの順位相関係数が高いモデルがあった場合，そのモデルは極めて汎用的であると言えるだろう．あるいは，全てのデータセットとはいかなくとも，複数のデータセットで高い性能のモデルがあった場合，そのうちいずれかのデータセットに向けてチューニングしたモデルを，他のデータセットにそのまま適用できる可能性がある．また，学習手法の異なるモデルの比較から，それぞれの学習手法の特性が明らかになるかもしれない．

3 句の分散表現計算モデルの学習

本研究では，図1に示したように，3種類の訓練データを用い，それぞれについて，句の分散表現を合成するモデルを学習する．具体的には，言い換え関係にあるbigram，ukWaC[1] から抽出した関係パタン，PPDBから抽出した句のペアについて，構成単語の分散表現を合成するニューラルネットワークを学習する．この，分散表現を合成するニューラルネットワークを，本研究ではエンコーダと呼称する．本節では，それぞれの訓練データを用いた学習手法，および，本研究で用いるエンコーダについて概説する．

3.1 学習手法

3.1.1 Bigram，PPDB の学習

Wieting[4] らの構築した，言い換え関係にあると考えられる Bigram，および，PPDB から抽出した句のペアを学習データとし，句の分散表現を合成するモデルを学習する．学習の目的としては，各エンコーダが，似た意味の句については似た分散表現を出力するように学習したい．つまり，言い換え関係にある句のペア $\langle p_1, p_2\rangle$ の集合を P としたとき，各ペア $\langle p_1, p_2\rangle \in P$ に含まれる句の分散表現同士が互いに似るように学習を行いたい．これを達成するために，ミニバッチを用いた AdaGrad [2] によって，次式を最小化する．

$$\begin{aligned}\sum_{\langle p_1,p_2\rangle\in P} &\max(0, \delta - h_{p_1}\cdot h_{p_2} + h_{p_1}\cdot h_{\overline{p_1}}) \\ &+\max(0, \delta - h_{p_1}\cdot h_{p_2} + h_{p_2}\cdot h_{\overline{p_2}}) \\ &+\lambda_\theta||\theta||^2 + \lambda_X||X_{initial} - X||^2. \qquad (1)\end{aligned}$$

ここで，λ_θ および λ_X は正則化項の重み係数，θ はエンコーダのパラメータ，X は語彙中の全単語の分散表現を表す行列，$X_{initial}$ は X の初期値，h_p はエンコーダで計算した p の分散表現，δ はマージン，$\overline{p}$ はミニバッチ内でサンプリングした負例である．すなわち，与えられた句のペアの分散表現の内積が，負例との内積よりもマージン以上に大きくなるように学習を行う．

負例については，与えられた句のペアを除き，ミニバッチ内で最も似ている句とする．例えば，ペア $\langle p_1, p_2\rangle$ について，以下のような式で表される $\overline{p_1}$ を負例とする．

$$\overline{p_1} = \mathrm{argmax}_{\overline{p_1}:\langle\overline{p_1},\cdot\rangle\in P_b\setminus\langle p_1,p_2\rangle} h_{p_1}\cdot h_{\overline{p_1}},$$

ここで，$P_b \subseteq P$ は現在のミニバッチとする．

ハイパーパラメータについても，Wieting ら [4] と同様に，初期学習率を単語の分散表現については 0.5，エンコーダについては 0.05 とし，エポック数は 5，マージンの値は 1 とした．さらに，λ_θ，λ_X，およびミニバッチのサイズについてはグリッドサーチを行った．具体的には，λ_θ は $\{10, 1, 10^{-1}, ..., 10^{-6}\}$，$\lambda_X$ は $\{10^{-1}, 10^{-2}, 10^{-3}, 0\}$，ミニバッチのサイズは $\{100, 250, 500, 1000, 2000\}$ の範囲で探索を行い，開発データで最も性能の良いものを採用した．

3.1.2 関係パタン

関係パタンについては，関係パタン間に同義関係を付与した訓練データが存在しないため，Takase ら [5] と同様，ラベルなしデータからの学習を行う．具体的には，ラベルなしデータからの単語の分散表現学習手法である，負例サンプリングを用いた Skip-gram モデルをエンコーダの学習に応用する．定式化すると，関係パタン p の対数尤度を次式で表す．

$$\sum_{\tau\in S_p}\left(\log\sigma(h_p^\top \tilde{x}_\tau) + \sum_{k=1}^{K}\log\sigma(-h_p^\top \tilde{x}_{\bar{\tau}})\right), \qquad (2)$$

ここで，S_p はコーパス中に出現した関係パタン p の周囲 L 単語の集合，K は負例サンプリングの数，$x_{\bar{\tau}}$ は負例 $w_{\bar{\tau}}$ の分散表現である．関係パタン抽出器（Reverb[6]）を ukWaC コーパスに適用して得られた関係パタンについて，SGD を用いて上記の対数尤度の最大化を行う．ハイパーパラメータについては，窓幅 $L = 5$，負例数 $K = 20$，サブサンプリングのパラメータを 10^{-5} とした．

なお，PPDB，関係パタンのどちらにおいても，単語の分散表現の次元は $d = 300$ とし，word2vec[2] に実装されている，負例サンプリングを用いた Skip-gram モデルを ukWaC に適用した結果で初期化した．

3.2 エンコーダ

3.2.1 加法構成

構成単語から句の分散表現を計算するにあたり，最も単純な手法として，構成単語の分散表現の平均を計算する，加法構成がある [9]．厳密には，計算対象の句が

[1] http://wacky.sslmit.unibo.it/doku.php?id=corpora

[2] https://code.google.com/archive/p/word2vec/

エンコーダ	形容詞–名詞	名詞–名詞	動詞–名詞	関係パタン	PPDB
加法構成（初期値）	0.42	0.40	0.52	0.25	0.28
Bigram で学習					
加法構成	**0.61**	0.41	0.53		
LSTM	0.59	0.45	0.52		
GRU	**0.61**	0.47	**0.57**	–	–
GAC	0.56	**0.48**	0.55		
CNN	0.60	0.40	0.40		
関係パタンを学習					
加法構成	0.43	0.40	0.51	0.31	0.25
LSTM	0.42	0.40	0.42	0.33	0.33
GRU	0.42	0.38	0.44	0.35	0.34
GAC	0.40	0.40	0.50	0.36	0.29
CNN	0.45	0.40	0.49	0.35	0.36
PPDB で学習					
加法構成	0.53	0.21	0.43	**0.38**	0.45
LSTM	0.51	0.10	0.26	0.26	0.40
GRU	0.55	0.15	0.34	0.26	0.39
GAC	0.48	0.39	0.45	0.36	**0.46**
CNN	0.40	0.34	0.28	0.19	0.27
PPDB+Bigram で学習					
加法構成	0.59	0.39	0.46	0.33	0.41
LSTM	0.58	0.20	0.30	0.26	0.43
GRU	0.55	0.33	0.33	0.26	0.42
GAC	0.57	0.38	0.42	0.32	0.45
CNN	0.47	0.36	0.31	0.22	0.25

表 1: 各データでのスピアマンの順位相関係数

T 個の単語から構成されている，すなわち，$w_1, ..., w_T$ と表せるとし，各単語 w_t の分散表現を $x_t \in \mathbb{R}^d$ とすると，$\frac{1}{T}\sum_{t=1}^{T} x_t$ という計算によって，この句の分散表現を得る．

3.2.2 Recurrent Neural Network 系の手法

近年，句や文の分散表現計算において，Recurrent Neural Network（RNN）およびその発展手法が成功をおさめている．RNN ベースの手法は，与えられた単語列に対し，順序に従って処理を行う．素朴な RNN では，勾配の消失／爆発問題があるため，学習時に長距離の依存関係を扱うのが難しい．この問題を解決するため，Hochreiter ら [8] は RNN にゲート機構を組み合わせた，Long Short-Term Memory（LSTM）を提案した．ゲート機構を利用し，LSTM に匹敵する手法しつつパラメータ数を抑えた手法として，Cho ら [1] は Gated Recurrent Unit（GRU）を提案した．さらに，Takase ら [5] は，ゲート機構が入力単語や隠れ状態の重要性を扱えると考え，ゲート機構と加法構成の組み合わせとして，Gated Additive Composition（GAC）を提案した．

3.2.3 Convolutional Neural Network

Convolutional Neural Network（CNN）は関係抽出や評判分析など，言語処理の様々なタスクで，句や文の意味をモデル化するために用いられている手法である．本研究では，句の分散表現を計算するために，次式のような，素朴な CNN を用いる．

$$\mathrm{conv_t} = \sigma(W_{conv}\left[x_{t-(n-1)/2}, ..., x_{t+(n-1)/2}\right] + b_{conv}), \tag{3}$$

$$\mathrm{pool} = \mathrm{maxpool}(\left[{}^{\mathrm{t}}\mathrm{conv}_1, ..., {}^{\mathrm{t}}\mathrm{conv}_{\mathrm{T}}\right]), \tag{4}$$

$$\mathrm{h} = g(\mathrm{pool}). \tag{5}$$

ここで，W_{conv} は $d \times (d * n)$ の行列，$b_{conv} \in \mathbb{R}^d$ はバイアス項，$\left[x_{t-(n-1)/2}, ..., x_{t+(n-1)/2}\right]$ は w_t の周囲 n 単語の分散表現を結合したもの，n はハイパーパラメータ（本研究では $n = 3$），maxpool は入力の行列の各行について，最も大きな値を抽出するという，最大値プーリングを表す．

4 実験結果

各データでの，各モデルのスピアマンの順位相関係数を表 1 に示した．表 1 の 2 段目は各 Bigram について学習したモデルの性能を示している．例えば，2 段目の形容詞–名詞の結果は，言い換え関係にある形容詞–名詞の句のペアを用い，節 3.1.1 で学習したモデルの性能を示している．ここで，各 Bigram に対して学習したモデルは，最上段に記した初期値，すなわち，ukWaC で学習した単語の分散表現に加法構成を適用して句の分散表現を計算した結果よりも，スピアマンの順位相関係数が上昇しており，各 Bigram の分散表現合成に適したモデルを学習できていると考えられる．

表 1 の 3 段目は，節 3.1.2 の手法で学習した，関係パタンの分散表現計算モデルの，各評価データでの性能を示している．加法構成については，Bigram，および，PPDB で初期値からあまり性能が変わっていない．Skip-gram モデルで加法構成のエンコーダを学習しようとした場合，各構成単語の分散表現の更新式は，Skip-gram モデルで単語の分散表現を学習する場合に等しい [3]．学習用のコーパスも初期値を得るために用いたものと同じため，本質的には初期値の分散表現から変化していないのだと考えられる．ニューラルネットワークを用いたエンコーダに目を向けると，PPDB では，全てのエンコーダが初期値より向上している．また，形容詞–名詞，名詞–名詞で全てのエンコーダが，動詞–名詞で GAC と CNN が初期値とほぼ同程度の性能を達成している．この結果は，ラベルなしデータで学習した，関係パタンの分散表現を合成するエンコーダは，関係パタンの分散表現計算にのみ特化している訳ではないと示唆している．

PPDB から抽出した句のペアを用いて学習したエンコーダの，各評価データでの性能を表 1 の 4 段目に示した．名詞–名詞および動詞–名詞では全てのエンコーダが，関係パタンでは加法構成と GAC 以外のエンコーダが初期値よりも悪化している．しかしながら，GAC については，名詞–名詞における GRU や LSTM，関係パタンにおける CNN のように，極端に低い値にはなっていない．このことから，節 3.1.1 の手法で学習した GAC は，任意の句に対して適用できる可能性がある．

表 2 に示したように，PPDB から構築した訓練データには，形容詞–名詞の bigram を含むペアが，名詞–名詞や動詞–名詞を含むペアよりも多い．表 2 の 4 段目，形容詞–名詞について，CNN 以外の全エンコーダが初期値よりも高い性能を達成しているのはこのためだと考えられる．また，図 2 に示したように，訓練データ量が増すに従い，関係パタンデータセットでの性能も向上する．では，訓練データをさらに増やした場合，性能はどのようになるのだろうか．

形容詞–名詞，名詞–名詞，動詞–名詞の訓練データを PPDB 内の句の訓練データに追加して学習した際の性能を表 1 の最下段に示す [4]．形容詞–名詞については，PPDB から抽出した句のペアのみで学習したモデル（4 段目）よりも性能が上がっているが，形容詞–名詞のみで学習したモデル（2 段目）よりも性能が低い．また，名詞–名詞，動詞–名詞については，初期値の性能よりも低下しており，関係パタンを対象に学習したエンコーダ（3 段目）よりも低い．さらに，関係パタンにおいても，性能に向上があまり見られない．このことから，Wieting[4] らの手法で学習したエンコーダは，Takase[5] らの手法でラベルなしデータから学習したエンコーダと比べ，汎用性は低いと考えられる．

[3]厳密には，関係パタンという句にまとめることで，窓幅が変化している．

[4]学習時のハイパーパラメータは PPDB で学習した際のものと同一とした．

句の種類	PPDB の訓練データ中で対象の句を含むペアの数
形容詞–名詞	8,419
名詞–名詞	3,323
動詞–名詞	3,686

表 2: PPDB の訓練データ（60,000 ペア）中で，対象の bigram を含むペアの数

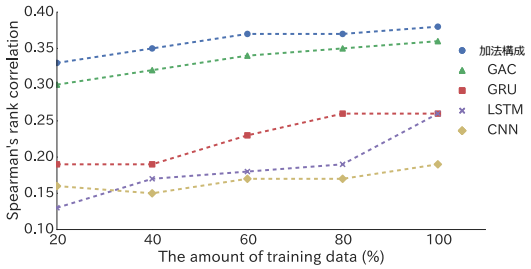

図 2: PPDB から抽出した句のペア（訓練データ）の量を変化させた際の関係パタンでの性能

5 おわりに

本研究では，句の分散表現を計算するエンコーダが，汎用的な合成関数を獲得できているかを調査した．実験の結果，ラベルなしデータから，関係パタンの分散表現の合成を学習したエンコーダは，PPDB で学習したものよりも汎用性が高いと示唆されており，単純な手法である，Skip-gram モデルで学習した単語の分散表現の加法構成でも，各評価データで安定的に高い性能を示した．今後は，文単位の分散表現を計算できるよう学習したモデルや，ニューラル翻訳モデルではどのような結果になるのか調査したい．

謝辞 本研究は文部科学省科研費 15H01702, 15H05318，および JST，CREST の支援を受けたものである．

参考文献

[1] K. Cho. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Procs. EMNLP2014*, pp. 1724–1734, 2014.

[2] J. Duchi. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, Vol. 12, pp. 2121–2159, 2011.

[3] J. Ganitkevitch et al. Ppdb: The paraphrase database. In *Procs. NAACL-HLT 2013*, pp. 758–764, 2013.

[4] J. Wieting et al. From paraphrase database to compositional paraphrase model and back. *TACL 2015*, Vol. 3, pp. 345–358, 2015.

[5] S. Takase et al. Composing distributed representations of relational patterns. In *Procs. ACL2016*, pp. 2276–2286, 2016.

[6] A. Fader. Identifying relations for open information extraction. In *Procs. EMNLP2011)*, pp. 1535–1545, 2011.

[7] G. Frege. *Die grundlagen der arithmetik*. 1884.

[8] S. Hochreiter. Long short-term memory. *Neural Computation*, Vol. 9, No. 8, pp. 1735–1780, 1997.

[9] J. Mitchell and M. Lapata. Composition in distributional models of semantics. *Cognitive Science*, Vol. 34, No. 8, pp. 1388–1439, 2010.