4A2-2

LSTMを用いた句のベクトル表現学習

Learning Distributional Representations of Phrases Using LSTM

水口凱 *1 Gai Mizuguchi　高谷智哉 *2 Tomoya Takatani　山田整 *2 Hitoshi Yamada　三輪誠 *1 Makoto Miwa　佐々木裕 *1 Yutaka Sasaki

*1豊田工業大学 Toyota Technological Institute　*2トヨタ自動車 Toyota Motor Corporation

Distributional representations of words are effective to natural language processing tasks. We have to represent not only word meaning but also phrase meaning to represent the meaning of broader regions of texts. To obtain distributional representations of phrases, we employ a model that obtains the characteristics of phrases and learns distributional representations of phrases using long short-term memory (LSTM). LSTM is a kind of neural network that can treat sequential data. We trained our model on British National Corpus (BNC) and evaluated our model on Grefenstette and Sadrzadeh's data set (GS'11) using the Spearman rank correlation coefficient. As a result, our model achieved the Spearman correlation of 0.523, which was competitive with the state-of-the-art method.

1. 背景と目的

近年，ニューラルネットワーク・深層学習技術の発展とともに，自然言語処理は大きな飛躍をみせている．このようなニューラルネットワークを用いる言語処理では，単語を数値ベクトルと対応付けた単語ベクトルを基盤として用いることが多い．単語ベクトルを獲得する手法は skip-gram[Mikolov 13] などがあり，このような手法によって得られた単語ベクトルを用いてベクトル空間上で単語間の類似度を測ることができ，“big” と “large”，“tall” と “high” のように意味の似た単語同士の類似度が大きくなることが示されている．このような単語ベクトルは，言語処理のさまざまなタスクにおいて大きな成果をあげている．

しかし，単語のみを表現する単語ベクトルだけでは句や文の意味をうまく表現できない．このような句や文の情報は，質問応答システムや機械翻訳，自動要約生成などの自然言語処理の応用タスクに重要であり，単語ベクトルよりもより大きなまとまりの意味を表現するベクトルの獲得が必要である．従来の句ベクトルの表現手法には，テンソルを用いた句の構成に基づく手法が多く提案されている．このような句は，単語の列として表現できるため，単語列のような系列データを表現できるニューラルネットワークである Long Short-Term Memory（LSTM）が有用であると考えられる．

本研究では，句の表現ベクトルの獲得を目指し，句の表現ベクトルを LSTM を用いて学習する手法を提案する．

2. 関連研究

2.1 単語の表現学習

単語の意味や単語間の関係をコンピュータ上で表現する手法として，単語の意味的な特徴をニューラルネットワークによって抽出することで，単語を数値ベクトルで表現した単語ベクトルを獲得するモデルが提案されている．単語ベクトルは同じような意味を持つ単語同士が近くなるだけでなく，単語の意味の演算を行うことができることが知られている．例えば，$v(\mathrm{w})$ を単語 w に対応する単語ベクトルすると，$v(king)-v(man)+v(woman) \approx v(queen)$ のような関係の演算が成り立つことが示されている．このような単語の意味情報を含んだ単語ベクトルは自然言語処理のタスクにおいて大きな成果をあげている．

2.1.1 skip-gram

skip-gram[Mikolov 13] は文章から単語ベクトルを学習するモデルである．skip-gram は文中のある単語に注目し，その単語の周辺（文脈）に存在する単語を予測することで単語ベクトルを学習している．このモデルは単語列 $w_1, w_2, \ldots, w_T$ が与えられたとき，次式の目的関数を最大化する．

$$J = \frac{1}{T}\sum_{t=1}^{T}\sum_{-c\le j\le c, j\neq 0} \log p(w_{t+j}|w_t) \tag{1}$$

ここで，c はウィンドウサイズである．また，$p(w_{t+j}|w_t)$ は単語から予測される周辺単語の確率であり，次式で定義される．

$$p(w_{t+j}|w_t) = \frac{\exp(c_{w_{t+j}} \cdot v_{w_t})}{\sum_{w'\in W}\exp(c_{w'} \cdot v_{w_t})} \tag{2}$$

ここで，v_w, c_w は単語 w が注目された場合，文脈に現れた場合それぞれに対応するベクトルであり，W は学習に用いた文章中の語彙である．式 (2) に対する勾配を求め，確率的勾配降下法 [Bottou 10] を用いることで単語ベクトル v_{w_t} と文脈ベクトル $c_{w_{t+j}}$ を更新する．

2.2 句のベクトル表現学習

句の意味的な特徴を捉えるために，句を数値ベクトルで表現する手法も研究されている [Kartsaklis 2012, Van de Cruys 13, 橋本 15].

Grefenstette ら [Grefenstette 11] は，動詞の表現は文中の主語と目的語の組み合わせによって決定されるという考えに基づき，動詞をテンソルとして表現する手法を提案した．この手法では，主語と目的語を同次元のベクトル $v(S), v(O)$ で表現し，そのクロネッカー積 $v(S)\otimes v(O)$ で主語と目的語の組み合わせを表現する．ある他動詞 V についてのテンソル表現 $M(V)$ は N 通りの主語・目的語の組み合わせで次のように計算する．

$$M(V) = \sum_{i=1}^{N} v(S_i)\otimes v(O_i) \tag{3}$$

連絡先: 水口凱，豊田工業大学
〒 468-8511 愛知県名古屋市天白区久方二丁目 12 番地 1

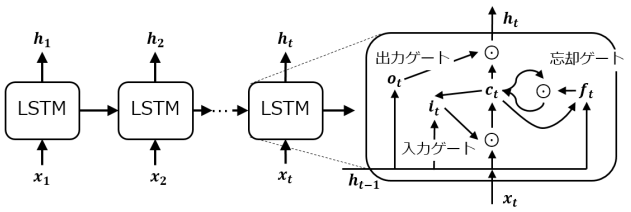

図 1: LSTM

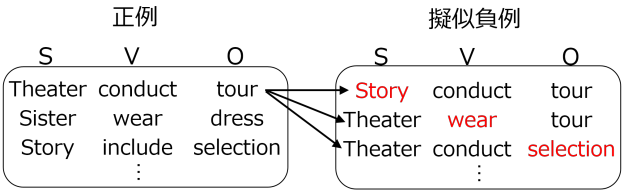

図 2: 擬似負例の作成

この Grefenstette らの手法に基づいて，Kartsaklis ら [Kartsaklis 2012] は主語・目的語を表現する単語ベクトルと，他動詞を表現するテンソルを組み合わせることで主語，動詞，目的語からなる句を表現するベクトルを計算する手法を提案した．

また，このテンソルで単語や句を表現するという考えに基づき，Van ら [Van de Cruys 13] は主語，動詞，目的語の 3 要素を 3 階のテンソルで表現する手法を提案した．この 3 階テンソルは，文章中に出現した（主語，動詞，目的語）の組み合わせの共起情報を考慮し，ある（主語，動詞，目的語）の組み合わせがどの程度尤もらしいかを示したものとなっている．Van らはこのテンソルを，主語を表現するベクトルの集合，動詞を表現する行列の集合，目的語を表現するベクトルの集合の 3 要素に分解した．この分解を行うことによって，主語・目的語の単語の表現ベクトルと，動詞の表現行列を獲得することができる．獲得した表現ベクトル・表現行列を用いて主語・動詞・目的語からなる句の表現が計算できることを示した．

さらに，この Van らの手法に基づいて，橋本ら [橋本 15] は動詞と主語・目的語の共起情報を持つ 3 階テンソル T が

$$T = P \times A_1 \times A_2 \tag{4}$$

という形でテンソル分解されるとした考えに基づく手法を提案した．P は動詞のパラメータを示す 3 階テンソルで，A_1, A_2 は主語・目的語に対するパラメータ行列である．P の各スライス $P(i)$ は各動詞を表す行列であり，A_1, A_2 の各列ベクトル a_1, a_2 は主語・目的語を表すベクトルである．ある特定の句の組 (i, j, k) が与えられたとき，その組の尤もらしさは以下の式で定義される．

$$T_{i,j,k} = a_1(j)^{\mathrm{T}} \mathrm{P}(i) a_2(k) \tag{5}$$

モデルパラメータは，尤もらしい組み合わせとそうでない組み合わせとの分類により学習する．学習に用いたそれぞれの句の組 (i, j, k) について，(i', j', k') を別の句の集合からランダムに獲得し，学習データ中に存在しない負例の組 (i', j, k)，(i, j', k)，(i, j, k') を作成し，次の誤差関数を定義する．

$$\begin{aligned} -\log \sigma(T_{i,j,k}) \quad & -\log(1 - \sigma(T_{i',j,k})) \\ & -\log(1 - \sigma(T_{i,j',k})) \\ & -\log(1 - \sigma(T_{i,j,k'})) \end{aligned} \tag{6}$$

この誤差関数の総和を目的関数として，これを AdaGrad を用いて最小化することでパラメータの最適化を行う．この最適化によって，同じような主語・目的語を持つ動詞を表す動詞の行列表現が近くなるように学習する．

2.3 LSTM

LSTM は，単語列などの系列データを扱えるニューラルネットワークの一種である．図 1 が LSTM の概略である．LSTM は内部にメモリを持ち，入力ベクトルによってメモリを更新し，更新したメモリの値から出力ベクトルを求めることで，入力した系列データの時系列関係を考慮することができる．時刻 t での入力 x_t が与えられたときの出力 h_t は以下の式で定義される．

$$i_t = \sigma(W^{(i)} x_t + U^{(i)} h_{t-1} + b^i), \tag{7}$$
$$f_t = \sigma(W^{(f)} x_t + U^{(f)} h_{t-1} + b^f), \tag{8}$$
$$o_t = \sigma(W^{(o)} x_t + U^{(o)} h_{t-1} + b^o), \tag{9}$$
$$u_t = \tanh(W^{(u)} x_t + U^{(u)} h_{t-1} + b^u), \tag{10}$$
$$c_t = i_t \odot u_t + f_t \odot c_{t-1}, \tag{11}$$
$$h_t = o_t \odot \tanh(c_t). \tag{12}$$

ここで，σ はシグモイド関数，W, U は重み行列，b はバイアスベクトル，c_t は t 番目の LSTM セル，$\odot$ はアダマール積，i_t は入力ゲートの出力，f_t は忘却ゲートの出力，o_t は出力ゲートの出力である．さらに，系列データを前後二方向から入力する双方向 LSTM[Graves 05] がある．

3. 提案手法

本手法では，LSTM を用いて句を表現し，句の尤もらしさを学習することで，主語 S，動詞 V，目的語 O からなる句のベクトル表現を獲得するモデルを提案する．今回考慮している句の尤もらしさとは，橋本らの手法と同様，句が文章中に現れるか否かである．提案モデルが対象とする句は，主語・動詞・目的語の 3 単語からなる句である．テキストや発話を大規模に集めたコーパスを構文解析することで主語・目的語が名詞である動詞を句として抽出する．抽出した句から学習用データを作成し，この学習用データを提案するモデルに入力することで，句のベクトル表現を獲得する．

3.1 擬似負例の作成

句のベクトル表現の学習に際し，文章中に現れる正例の句のみだけでなく，学習データ中に存在しない擬似負例の句を用いて学習を行う．擬似負例は橋本らの方法と同様に作成する（図 2）．

3.2 双方向 LSTM を用いた学習

提案手法のモデルを図 3 に示す．図 3 に示したモデルでは，句を双方向 LSTM の入力とし，双方向 LSTM の各セルの出力平均を concatenate した $\overline{h_i}$ を獲得する．双方向 LSTM の出力を平均することで主語・動詞・目的語の特徴を持ったベクトルを作成する．この $\overline{h_i}$ を入力とした多層ニューラルネットワークを利用して，句ベクトル p_i を得る．

この p_i の最適化のために，入力した正例・擬似負例の句に対し，多層ニューラルネットワークの出力 o_i について，正例か擬似負例か，の二値分類を行う．この分類を行うことによって，動詞と主語・目的語の関係を考慮することができる．例え

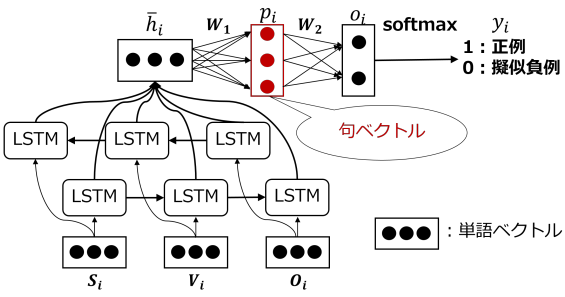

図 3: 提案モデル

ば，“conduct” という動詞は “theater” や “tour” といった単語と句を作る事が多いが，“story” や “selection” といった単語とは句を作らない傾向にあるということを学習すると期待できる．学習では，以下の目的関数 J を最小化するように各パラメータを更新する．

$$J = \Sigma_i - \widetilde{y_i} \log(y_i) - (1 - \widetilde{y_i}) \log(1 - y_i) \quad (13)$$

$$y_i = \mathrm{argmax}_c P(C) \quad (14)$$

$$P(C) = \mathrm{softmax}(o_i) \quad (15)$$

$$o_i = f(W_2 p_i + b_2) \quad (16)$$

$$p_i = f(W_1 \overline{h}_i + b_1) \quad (17)$$

ここで，$\widetilde{y_i}$ は正解ラベル，y_i はモデルが予測したラベル，W_1, W_2 は重み行列，b_1, b_2 はバイアスベクトル，$\mathrm{softmax}(x)$ は入力 x に対するソフトマックス関数，f は Rectified Linear Unit（ReLU）である．

4. 実験

4.1 学習データ

句の尤もらしさを学習するためのコーパスとして British National Corpus（BNC）を用いた．BNC は総語彙数が約 1 億語のコーパスである．BNC は文語と口語の文章を両方を含んでおり，約 9 割が文語，約 1 割が口語となっている．文語の文章には，さまざまな地域の新聞記事，学術雑誌，出版された書籍などを含んでいる．口語の文章には，スピーチの書き起こしなどさまざまな状況で収集された会話を含んでいる．構文解析器 enju を用いて主語と目的語が名詞である動詞句（1,767,366 句）を獲得し，そのうち 10,000 句を開発データ，残りを学習データとして用いた．開発データでは固定した擬似負例（10,000 句）を事前に用意した．

4.2 評価方法

開発データでの評価には正例と擬似負例における分類の正解率を利用した．

表 1: GS’11 の例文

句	類似度スコア
man provide money man supply money	7
judge try action judge test action	4
employee buy property employee bribe property	1

獲得した句のベクトル表現の妥当性の評価には，Grefenstette と Sadrzadeh のデータセット（GS’11）[Grefenstette 11] を用いた．GS’11 は，二つの他動詞が同一の主語と目的語を伴うとき，どの程度の意味的な類似度があるかを 200 事例に関して人手によって 1 から 7 の七段階のスコア付けを行ったデータセットである．データセット中の句のペアの一部を表 1 に示す．各事例について複数人がスコアを与えており，本稿ではその平均値をとっている．GS’11 の類似度スコアと，学習したモデルによって得られる GS’11 中の句のペアの cos 類似度とをスピアマンの順位相関係数で評価した．スピアマンの順位相関係数は 2 データ間の関係が任意の単調関数によってどの程度表現できるかを評価する指標である．データ X，Y 間のスピアマンの順位相関係数 ρ は，$-1 \leq \rho \leq 1$ の範囲で値をとり，ρ が 1 に近いほど正の相関が，ρ が -1 に近いほど負の相関があると言える．

4.3 パラメータ

本実験で用いたパラメータを表 2 に示す．単語ベクトルは word2vec の skip-gram モデルにて事前学習を行った数値で初期化した．単語ベクトルの事前学習に用いたコーパスは BNC である．また，多層パーセプトロン中の重み W_1, W_2 は乱数で初期化した．学習には Adam[Kingma 2015] を用いた．学習時のハイパーパラメータである，学習率，L2 正則化の係数，単語ベクトルの次元数は，開発データにおける分類精度を用いてパラメータチューニングを行った．また，双方向 LSTM の出力の利用方法の妥当性を評価するため，双方向 LSTM の各セルの出力平均を concatenate したものでなく，順方向・逆方向の最後の出力を concatenate して二値分類を行うモデルでも評価した．

5. 結果

パラメータを変えずにモデルを変更した場合，単語ベクトルの次元を変更した場合，LSTM の出力を変更した場合の二値分類の正解率を図 4，5，6 に示す．それぞれ，パラメータを変えずにモデルを変更したもの，単語ベクトル図 4 から，双方向 LSTM での分類正解率 (bi-LSTM_hidden_layer_1) が LSTM での精度 (LSTM_hidden_layer_1) より高くなった．さらに，事前に学習した単語ベクトルを利用した場合 (bi-LSTM_hidden_layer_1)，利用しない場合 (bi-LSTM_hidden_layer_1(no_pretrain)) に比べて，収束までの時間が短くなり，性能が向上した．多層パーセプトロンについては，利用しない場合 (hidden_layer_0) や中間層の数を 2，全体の層の数を 4 とした場合 (hidden_layer_2) に比べて，層の数を 3 としたときが収束速度が速く分類精度も高くなった．図 5 に示した通り単語ベクトルの次元数は 600 次元としたときが最もよく分類できた．また，図 6 より LSTM からの出力

表 2: 各種パラメータ

パラメータ	値
単語ベクトル次元	600
双方向 LSTM 出力次元	600×2
モデルの重み W_1 次元	1200×600
モデルの重み W_2 次元	600×2
学習率	0.001
L2 正則化の係数	0.0001
ミニバッチサイズ	10,000

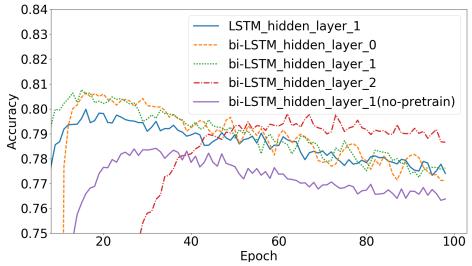

図 4: 二値分類の正解率（モデル）

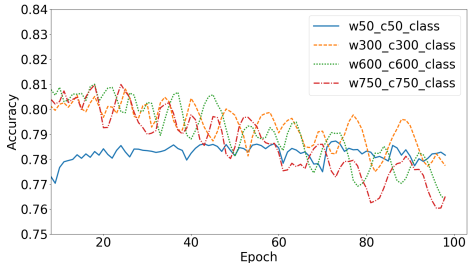

図 5: 二値分類の正解率（次元）

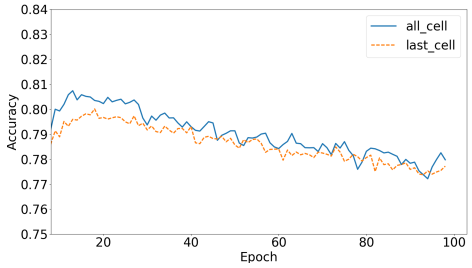

図 6: 二値分類の正解率（LSTM 出力）

は全てのセル出力を平均したもの (all_cell) が LSTM の最後の出力を用いたもの (last_cell) に比べてよい結果となった．

次に提案手法と橋本らの手法 [橋本 15] をスピアマン順位相関係数で評価した結果を表 3 に示す．提案手法により，従来手法に近いスピアマン順位相関係数の値を得ることができた．また表 4 に GS'11 中の人手で付けられたスコア順位と，獲得した句のベクトル表現の cos 類似度の一部を示す．表 4 の “writer write book” と “writer spell book”，“man write song” と “man spell song” それぞれの cos 類似度順位が低い順位となった．これより，提案手法は主語と目的語の違いによる 2 つの他動詞の違いを十分に捉えられていない場合があることがわかる．

6. まとめ

本研究では句の情報をよく表す句の表現ベクトル獲得を目指し，LSTM を用いて句を表現する手法を提案した．提案した手法では実際の文章中に現れる正例の句と正例中の一単語をランダムに置き換えて作成した擬似負例の句を分類することによって，句のベクトル表現を学習した．BNC を用いて学習し，GS'11 において評価した結果，スピアマン順位相関係数値が 0.523 となり，従来手法に近い結果を得ることができた．今後は異なる構成要素による句の違いをより正確に表現できるようにモデルを拡張する予定である．

表 3: GS'11 における評価

手法	相関値
提案手法	0.523
橋本ら [橋本 15]	0.552

表 4: 順位例

句	GS'11 スコア	GS'11 順位	cos 類似度 類似度	cos 類似度 順位
report draw attention report attract attention	6.5	17	0.999	1
writer write book writer spell book	153	2.0	0.704	148
man write song man spell song	10	6.7	0.716	144

参考文献

[Mikolov 13] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In *Proceedings of Workshop at the International Conference on Learning Representations*, 2013.

[Bottou 10] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of the 19th International Conference on Computational Statistics*, 2010.

[Graves 05] Alex Graves and Jurgen Schmidhuber. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 2005.

[橋本 15] 橋本和真，鶴岡慶雅．テンソル分解に基づく述語項構造のモデル化と動詞句の表現ベクトルの学習．第 21 回言語処理学会年次大会，2015.

[Grefenstette 11] Edward Grefenstette and Mehrnoosh Sadradeh. Experimental Support for a Categorical Compositional Distributional Model of Meaning. In *Proceedings of the* 2011 *Conference on Empirical Methods in Natural Language Processing*, 2011.

[Van de Cruys 13] Tim Van de Cruys, Thierry Poibeau, and Anna Korhonen. A Tensor-based Factorization Model of Semantic Compositionality. In *Proceedings of the* 2013 *Conference of the North American Chapter of the Association for Computational Linguistics* : *Human Language Technologies*, 2013.

[Kartsaklis 2012] Dimitri Kartsaklis, Mehrnoosh Sadrzadeh, and Stephen Pulman. A Unified Sentence Space for Categorical Distributional-Compositional Semantics: Theory and Experiments. In *Proceedings of the 24th International Conference on Computational Linguistics*, 2012.

[Kingma 2015] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference for Learning Representations*, 2015.