2F5-OS-01b-2

単語間の距離を考慮した単語の意味表現の学習手法

Learning Word Representation by taking account of Distances between Words

伊藤 誠[*1]
Makoto Ito

高木 友博[*1]
Tomohiro Takagi

[*1] 明治大学理工学研究科基礎理工学専攻
Computer Science Course, Graduate School of Science and Technology, Meiji University

Recent models for leaning word representation have succeeded in capturing high-quality semantic information as vectors. In this paper, we present the feature values based on distances between word to word and the model that is extended the Matrix Factorization technique in several ways. These two elements improve results on word similarity tasks.

1. はじめに

近年，単語の意味的な情報をベクトル空間で表す研究が注目されている．一般に学習手法は，単語間の共起情報に基づいており，代表的な手法として Word2Vec[Mikolov 13]が挙げられ，その有用性も示されている[Baroni 14]．

その一方で，行列因子分解，または特異値分解を応用して，Word2Vec よりも高い精度が得られる手法として GloVe[Pennington 14]，SPPMI[Levy 14]も提案されている．

本稿では，単語間の距離を考慮した特徴量と，その特徴量を適応させるための行列因子分解を拡張したモデルを提案する．さらに，WordSim353 をはじめ，単語ペアとその類似度が記述されたデータセットを用いて実験を行い，提案手法が先行研究よりも高い数値が得られることを報告する．

2. 行列因子分解

行列因子分解[Koren 09]は，与えられた$I \times J$の行列Xに対して，より低ランクな$I \times K$の行列Vと$K \times J$の行列Uに分解するモデルである．Kは基底の数であり，事前に決めておく．図 1 にその概要を表す．図 1 で示すように$v_1 = (v_{11}, v_{12}, \dots v_{1K})^{\mathrm{T}}$と$u_1 = (u_{11}, u_{12}, \dots u_{1K})^{\mathrm{T}}$の内積${v_1}^{\mathrm{T}}u_1$が$x_{11}$に対応しており，その値が等しくなるように学習を行う．このパラメータV，Uを求めるための最も一般的なモデルとして，式(1)の目的関数Jが挙げられ，この二乗誤差関数の最小化問題を解く．

$$J = \sum_{(i,j)\in D} \left(x_{ij} - {v_i}^{\mathrm{T}} u_j\right)^2 + \lambda\left(\|v_i\|^2 + \left\|u_j\right\|^2\right) \qquad (1)$$

x_{ij}はXのi行j列の成分で，v_i，u_jはそれぞれVのi行目，Uのj列目のベクトルであり，Dは行列Xの成分x_{ij}において，欠損値を除く(i,j)のペアの集合である．式(1)には，過学習を防ぐために正則化項が加えられている．

また，式(1)の最小化問題の解法として大規模データにも対応できる確率的勾配降下法がよく用いられる．具体的には，データを 1 点ずつ，つまりx_{ij}毎にパラメータv_i，u_jの勾配を求め，更新していく．したがって，T回目の更新後のパラメータをv_i^T，u_j^Tとし，その時点でのJに関するそれぞれの勾配を$\frac{\partial J}{\partial v_i^T} = g\left(v_i^T\right)$，$\frac{\partial J}{\partial u_j^T} = g\left(u_j^T\right)$とおくと更新式は以下の通りになる．

連絡先：伊藤 誠，明治大学大学院ウェブサイエンス研究室，
〒214-0034 神奈川県川崎市多摩区三田 2-3242，
Email: ce46004@meiji.ac.jp

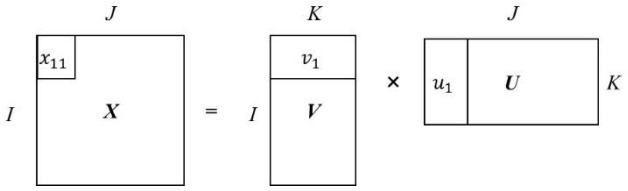

図 1 行列因子分解

$$v_i^T = v_i^{T-1} - \alpha g\left(v_i^T\right) \qquad (2)$$

$$u_j^T = u_j^{T-1} - \alpha g\left(u_j^T\right) \qquad (3)$$

αは学習率であり，事前に設定しておく．

3. 提案手法

提案手法の考え方は，分布仮説[Sahlgren 08][Schütze 97]に基づいている．概要を簡潔に述べると，「似た文脈，または同じ文脈に出現する単語同士は似た意味を持つ傾向にある」という仮説である．提案手法ではこの仮説から，ある単語対において近い位置での共起頻度が高ければ高いほど，似たベクトルになるように学習するモデルを構築した．この際，共起頻度の他に単語間の距離にも着目した．遠くに位置する単語よりも近接している単語の方が情報として価値があると考えられるので，特徴量にその情報を組み込んだ．

3.1 単語間の距離を用いた特徴量

特徴量には共起頻度，および単語間の距離を用いる．単語i，単語jにおける特徴量$x_{ij}(> 0)$は以下の式(4)の通りである．

$$x_{ij} = \frac{freq(i,j)}{\left(\frac{\sum_l dis_l(i,j)}{freq(i,j)}\right)} = \frac{freq(i,j)^2}{\sum_l dis_l(i,j)} \qquad (4)$$

$freq(i,j)$は単語iと単語jの共起頻度である．$dis_l(i,j)$は単語iと単語jのl回目$\left(1 \le l \le freq(i,j)\right)$の共起における単語間距離である．例として，”I put some bread in the toaster.”という文があった場合，$dis(bread, toaster) = 3$である．

このx_{ij}の値は，共起頻度が高ければ高いほど，また，共起した際に平均的に近接していればいるほど大きい値をとるようになる．

表 1 スピアマンの順位相関係数(×100)による比較

手法	WS353	RC	Mturk	MC
Basic MF(共起頻度)	26.34	12.37	25.49	22.61
SPPMI	34.44	20.14	25.01	16.33
GloVe	42.89	33.45	**42.40**	31.59
提案手法 ウエイト関数なし	40.35	23.61	32.31	32.93
提案手法 ウエイト関数(k=1)	42.41	37.33	36.97	41.54
提案手法 ウエイト関数(k=0.5)	**43.13**	**38.77**	37.97	**42.82**

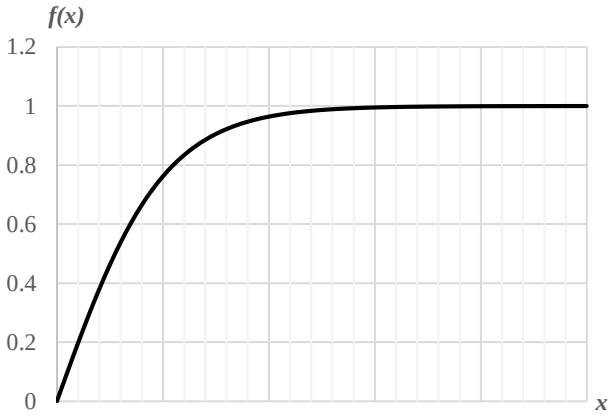

図2 *f(x)=tanh(x)*のグラフ(*x≥0*)

3.2 モデル

提案手法では，特徴量として式(4)のx_{ij}を用い，行列因子分解をベースにし，拡張を加えている．目的関数Jは式(5)の通りになっている．

$$J = \sum_{(i,j)\in D} f(x_{ij})\{log(x_{ij}+1) - \mu - b_i - c_j - v_i{}^{\mathrm{T}} u_j\}^2 \quad (5)$$

$$f(x) = tanh(kx) \quad (6)$$

μは全体のバイアスパラメータであり，b_i，c_jはそれぞれv_i，u_jにかかるバイアスパラメータである．また，$f(x)$はウエイト関数である．$k=1$の場合の$f(x)$は図 2 の様なグラフ形をしており，ある単語対におけるx_{ij}の値が低い場合に，更新時，互いの単語ベクトルへの影響を減らす目的で付与した．

このモデルにおいてv_i，u_jを学習する際には，x_{ij}の値が大きければ大きいほどv_iとu_jのベクトルの要素は似たような数値で表されるようになる．

3.3 最適化

提案手法におけるパラメータの最適化には，確率的勾配降下法ではなく，AdaGrad[Duchi 11]を用いた．したがって，T回目の更新後のパラメータv_i^T，u_j^Tは以下の通りである．

$$v_i^T = v_i^{T-1} - \frac{\alpha}{\sqrt{1+\sum_t^T g(v_i^t)^2}} g(v_i^T) \quad (7)$$

$$u_j^T = u_j^{T-1} - \frac{\alpha}{\sqrt{1+\sum_t^T g(u_j^t)^2}} g(u_j^T) \quad (8)$$

さらに，バイアスパラメータμ，b_i，c_jも同様にして更新を行っていく．

ここで，分母の勾配の二乗和に 1 を加えてから平方根をとっているが，それは勾配の二乗和の値が小さくなりすぎて学習率が大きくなりパラメータが発散してしまうのを防ぐためである[橋本 14]．

4. 実験

4.1 学習用データ

学習用データには，Reuters Corpus [1] を使用した．総記事数は 806,791 記事であり，その中で出現頻度が 50 よりも大きい単語を実験に使用した．その場合の総単語数は 52,716 語である．

4.2 検証用データと評価

検証用データには，WordSim353(WS353)，Rubenstein and Goodenough(RC)，MTurk，Miller and Charles(MC)の 4 つを用いた．これらは，ある単語対とその意味的な類似度のスコアが人手で付けられ，リスト化されたデータセットである．一例として MS353 には<(money，dollar)，8.42>といったデータが含まれている．

また，評価は，検証用データと同じ単語対における同様なリストをシステムで求めた単語ベクトルからコサイン類似度を用いて生成し，スピアマンの順位相関係数を求めることによって行った．

4.3 実験設定

全ての手法で共通して，基底の数は$K=100$，共起は文脈窓$window=10$，イテレーション回数は$T=500$に設定した．さらに，提案手法の設定に関して，初期学習率を$\alpha=0.05$，ウエイト関数$f(x)$をなしに，またはその係数を$k=1$，0.5に設定したものを用いて実験を行った．

評価は，$V+U$で表される行列から該当する単語ベクトルを抽出したものを用いて行う．

4.4 結果

表 1 に 4 つのデータセットに対してのスピアマンの順位相関係数を示す．本実験では，提案手法に加えて，比較手法として先行研究の GloVe，SPPMI を用意した．特に前者では，特徴量に単語間の共起頻度を用い，行列因子分解を拡張させたモデルが提案されており，本研究と類似している．後者に関して

[1] http://trec.nist.gov/data/reuters/reuters.html

Levy らは, SPPMI の値を特徴量とし, 特異値分解によって単語ベクトルを得ていたが, 本実験では, 式(1)で表される行列因子分解(ただし, $\lambda = 0$)を用いて単語ベクトルを求めるようにした. さらに, SPPMI と同様に共起頻度$x_{ij} = freq(i, j)$を特徴量に設定したもの(Basic MF)も用意する.

WS353, RC, MC の 3 つのデータセットでは, 提案手法の中でウエイト関数の係数 k=0.5 が最も良い相関値(43.28, 36.55, 40.07)が得られた.

3 つの比較手法(Basic MF, SPPMI, GloVe)は, どれも特徴量に単語の共起頻度のみをベースにしたものだが, 提案手法では, それに単語間の距離という情報を組み合わせることで相関値の向上が達成できたことから, 単語間の距離も特徴量を構築する際の重要な要因になることがわかる.

ウエイト関数を組み込んでいることも良い結果が得られた要因の 1 つであると考えられる. 実際に GloVe でも共起頻度が低い単語対に関して, 更新時に互いの単語ベクトルへの影響を減らすための関数が導入されている.

しかし, ウエイト関数の係数の値は調節が必要で, 本実験でも実験的に求まった係数である. したがって, 汎用性を満たすために, この係数を自動的に求まるような手法が必要である.

5. 結論

本稿では, 単語間の距離を考慮した特徴量を用いた行列因子分解を拡張したモデルの提案した. 実験の結果, 単語間の意味的な類似度を測るタスクに関して, 先行研究よりも過半数以上の検証用データセットに対して有用であることを示せた.

したがって, 単語の意味構造をベクトルで表す際に, 単語間の距離も重要な要因になることが期待できる.

参考文献

[Baroni 2014] Marco Baroni, Georgiana Dinu, and Germán Kruszewski: Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors, ACL, 2014.

[Duchi 2011] John Duchi, Elad Hazan, and Yoram Singer: Adaptive Subgradient Methods for Online Learning and Stochastic Optimization, JMLR, 2011.

[Koren 2009] Yehuda Koren, Robert Bell, and Chris Volinsky: Matrix Factorization Techniques for Recommender Systems, Journal Computer, 2009.

[Levy 2014] Omer Levy, and Yoav Goldberg: Neural Word Embedding as Implicit Matrix Factorization, NIPS, 2014.

[Mikolov 2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean: Distributed Representations of Words and Phrases and their Compositionality, NIPS, 2013.

[Pennington 2014] Jeffrey Pennington, Richard Socher, and Christopher D. Manning: GloVe: Global Vectors for Word Representation, EMNLP, 2014.

[Sahlgren 2008] Magnus Sahlgren: The distributional hypothesis, Italian Journal of Linguistics, 2008.

[Schütze 1997] Hinrich Schütze, and Jan O. Pedersen: A Cooccurrence-Based Thesaurus and Two Application to Information Retrieval , Information Processing and Management, 1997.

[橋本 14] 橋本 和真, 三輪 誠, 鶴岡 慶雅, 近山 隆: 述語構造に基づくニューラルネットワーク言語モデルの学習, 言語処理学会, 2014.