Deep Belief Network を用いた関連語・周辺語からの検索用語の予測

谷河 息吹[†]　馬 青[†]　村田 真樹[‡]
[†]龍谷大学大学院理工学研究科数理情報学専攻
[‡]鳥取大学大学院工学研究科情報エレクトロニクス専攻

1　はじめに

Google 等の検索エンジンは非常に精度が良くなってきており，それにともなって我々の生活の一部として欠かせない存在となってきている．しかしながら，検索エンジンを有効に使うには正しい用語で検索することが必要になってくるが，それは容易なことではなく，関連語・周辺語から正しい検索用語に自動で変換できることが望ましい（例えば,「テープ」,「記憶媒体」,「映像」から「VHS」）．

本研究では，関連語・周辺語またはそれらの語から構成される文を入力とし，機械学習を用いて正しい検索用語を提示するシステムを最終的な目標としているが，その第一歩として，提示される検索用語を計算機用語 10 語（「CPU」,「グラフィックボード」,「ハードディスク」,「メインメモリ」,「マザーボード」,「OS」,「光学ドライブ」,「PC ケース」,「電源ユニット」,「SSD]）に限定することにより，エキスパートシステムのような特定分野に特化したシステムの開発を行った．また，近年各分野で非常に良い結果を残している Deep Learning を使用し，Multi Layer Perceptron（以降略して「MLP」と呼ぶ）との比較を行うことにより，有効性を確認する．

機械学習では，教師データ・出力データをラベルと呼ぶため，本稿でも以下では検索用語をラベルと呼ぶこととする．

2　関連語・周辺語コーパス

機械学習手法を用いて関連語・周辺語からラベルを提示する場合，その学習データとして関連語・周辺語が多く含まれる言語コーパスが必要になってくる．本研究では，ラベルを説明している文書には関連語・周辺語が多く含まれると考え，インターネット上のウェブページを収集し，それを関連語・周辺語コーパスとして用いることとした．

2.1　手動収集データ・自動収集データ

ウェブページを収集する方法は，大別して手動収集と自動収集に分けられる．手動収集は，人が目視でラベルの説明文書か判断し，収集する方法であるのに対し，自動収集は，ラベルの後に「とは」・「は」・「というものは」・「については」・「の意味は」の 5 語を付けて（例：”CPU とは”，”SSD というものは”）検索エンジン Google で検索したものを説明文書と判断し，収集する方法である．以降略して，手動で収集したデータを「手動収集データ」，自動で収集したデータを「自動収集データ」と呼ぶ．自動収集の場合，5 語を付けて構成したクエリで検索するので収集した文書の中に重複したものがでてくるため，予め削除する[1]．

2.2　擬似データ

汎化能力を向上させるためには，手動収集データのような小規模なデータ以外に自動収集データのような大規模で適度なノイズがあるようなデータが必要になってくる．しかし，自動収集データは手動収集データと違ってラベルが不正確な可能性がある．そのため，手動収集データに対して，10 %の割合で欠損またはノイズまたはその両方を加えたデータ（以降略して「擬似データ」と呼ぶ）を作成し，学習データとして有効かどうかの確認を行う．擬似データの例を表 1 に示す（ノイズとして加えられた単語には，下線が引いてある）．

表 1: 擬似データの例

ラベル	説明文
CPU	クロック，コア，周波数，回路，…
CPU	内部，中央演算処理装置，頭脳，プロセッサ，ビット，…
CPU	周波数，ソフトウェア，コア，入力装置，演算，実行，…
⋮	⋮

2.3　手動ラベルデータ

自動収集データは，ラベルが正確とは限らないため，評価データとして用いても適切な評価とならない可能性がある．そのため，評価データとして自動収集データの中からラベルの正しいものを人手で選別し，そのデータ（以降略して「手動ラベルデータ」と呼ぶ）を用いることとした．

[1]fdupes コマンドを用いると簡単に行える．

2.4 ベクトル変換

機械学習に関連語・周辺語コーパスを用いる場合，ベクトルに変換する必要がある．テキストをベクトルにする主な方法として，単語ベクトル，N-gram ベクトルが挙げられる．N-gram ベクトルの場合，次元の数が膨大となり，機械学習で処理するには，非常に時間がかかるため単語ベクトルの方法を取ることとする．しかしながら，単語ベクトルの場合でも自動収集データの単語を含めると，次元の数が膨大となるため，手動収集データからベクトルを構成することとする．手順としては次のようにもととなるベクトルの構成を行っている．1. 各手動収集データの形態素解析を行い，名詞（固有名詞，サ変接続，一般）[2]を抽出する．2. 名詞が連続しているのであれば，日本語なら結合，英語なら空白を間に入れて結合し（名詞句の作成），ベクトルの構成要素とする．3. 構成したベクトルから 2 種類以上のラベルに共通するような語を除外する．各データをベクトルに実際に変換する際には，頻度ベクトルではなく，2 値ベクトルを用いることとする．これは頻度ベクトルでは，データの数が多くないと機械学習で正確な学習ができないためである．ベクトルに変換する際の問題点として表記揺れが挙げられるが，本稿では全角・半角の統一のみ行なっている．

3 Deep Learning

Deep Learning とは，従来の機械学習より深い層構造をしている機械学習手法全般のことを指す．以下では，Deep Learning の代表的な手法である Deep Belief Network（以降略して「DBN」と呼ぶ）とそれに用いられる Restricted Boltzmann Machine（以降略して「RBM」と呼ぶ）について述べる．

3.1 Restricted Boltzmann Machine

RBM とは，本来の Boltzmann Machine の入力層・隠れ層のユニット間の結合を制限することで計算を効率的にした確率的なニューラルネットワークである [1].

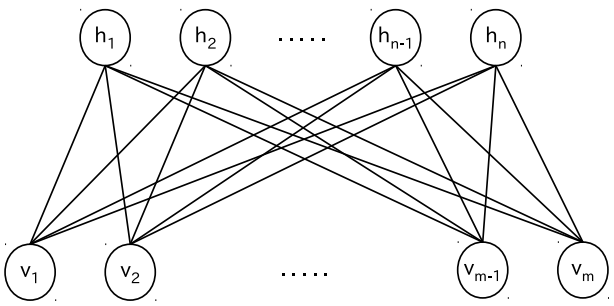

図 1: Restricted Boltzmann Machine

[2]サ変接続，一般の名詞を抽出するようにしているのは，名詞句の構成要素とするため．

RBM では以下の条件付き確率に従って，サンプリングを行うことで特徴抽出が行われる．

$$P(\mathrm{h}_i = 1|\mathbf{v}) = \mathrm{sigmoid}(\sum_{j=1}^{m} \mathrm{W}_{ij}\mathrm{v}_j + \mathrm{c}_i) \quad (1)$$

$$P(\mathrm{v}_j = 1|\mathbf{h}) = \mathrm{sigmoid}(\sum_{i=1}^{n} \mathrm{W}_{ij}\mathrm{h}_i + \mathrm{b}_j) \quad (2)$$

ただし，v_i は入力層のユニット，h_i は隠れ層のユニット，W_{ij} は入力層・隠れ層間の結合荷重，b_i は入力層のバイアス，c_i は隠れ層のバイアスである．

また，以下の更新式に従い，結合荷重・バイアスの更新が行われる．

$$\mathrm{W} \leftarrow \mathrm{W} + \epsilon(\mathbf{h}^{(1)}\mathbf{v}^T - P(\mathbf{h}^{(k+1)} = 1|\mathbf{v}^{(k)})\mathbf{v}^{(k)}) \quad (3)$$

$$\mathbf{b} \leftarrow \mathbf{b} + \epsilon(\mathbf{v} - \mathbf{v}^{(k)}) \quad (4)$$

$$\mathbf{c} \leftarrow \mathbf{c} + \epsilon(\mathbf{h}^{(1)} - P(\mathbf{h}^{(k+1)} = 1|\mathbf{v}^{(k)})) \quad (5)$$

ただし，ϵ は学習率，k はサンプリングの回数である．

学習が進むと入力層のサンプル（$\mathbf{v}^{(k)}$）[3]は，入力データ（$\mathbf{v}$）に近いものとなっていく．また，(1)，(2) の条件付き確率に従って，サンプリングを十分に行うことを Gibbs sampling というが，計算量が大きくなり実用的ではない．そのため (3)，(4)，(5) のようにサンプリングを k 回で打ち切る k-step Contrastive Divergence という手法が用いられることが多い．経験的に k の値は，1 の場合であっても良い結果が得られることが知られている [2] ため，本稿でも k の値は 1 を用いることとする．

3.2 Deep Belief Network

DBN は，本来経験則で行なっていた特徴を抽出する過程を機械学習の一連の動作の中に組み込もうとしてできたものであり，RBM を複数並べたものを特徴抽出器として利用する多層のニューラルネットワークである．また，教師なし学習である特徴抽出器の部分は Pre-training，教師あり学習である最終層の部分は Fine-tuning と呼ばれる．Fine-tuning は，教師あり学習であれば何でも良い．

DBN の学習は図 2 の場合，次のように行われる．1. すべての学習データを入力として，RBM1 のパラメータ（結合荷重，バイアス）を (3)，(4)，(5) の更新式に従い，予め決めた回数 (Epoch) だけ更新を行う．2. RBM1 のパラメータを固定し，RBM1 の隠れ層のサンプルを入力として，RBM2 のパラメータを予め決めた回数だけ更新を行う．3. RBM2 のパラメータを固定し，RBM2 の隠れ層のサンプルを入力として，RBM3 のパラメータを予め決めた回数だけ更新を行う．4. RBM3 のパラメータを固定し，RBM3 の隠れ層のサンプルを入力として，教師あり学習のモデルで学習を行う．こ

[3](1)，(2) の条件付き確率に従って，生成したデータをサンプルという．

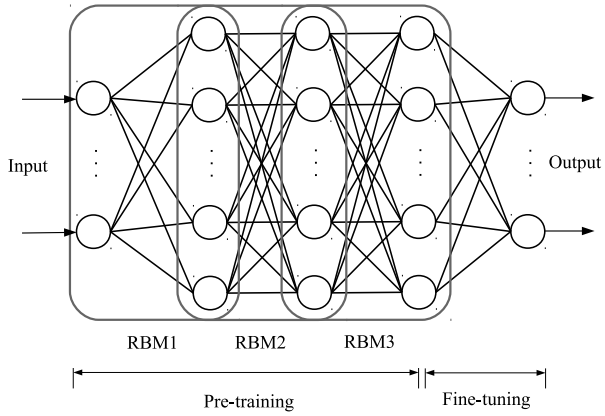

図 2: Deep Belief Network

のように，RBM の各層毎にパラメータを固定し，次層の RBM の学習を行うことを Greedy layer-wise という．説明のため，ここでは RBM の層の数を 3 としているが，それ以外でもよい．

4 実験

4.1 実験条件

学習データとして，手動収集データ 300 件を基本に自動収集データ 300 件，600 件，1,200 件，2,400 件を加えた場合，擬似データ 300 件，600 件，1,200 件，2,400 件を加えた場合，自動収集データと擬似データ 300 件+300 件，600 件+600 件，1,200 件+1,200 件，2,400 件+2,400 件を加えた場合に分け，効果の有無を見ることとする．評価データとしては，学習データと異なる自動収集データ（手動ラベルデータ 100 件）を用いることとする．2.4 節に従い，構成したベクトル（182 次元）をもとに学習データと評価データのベクトル変換を行う．MLP と DBN で 5-fold cross validation を行い，グリッドサーチをして学習データ毎の最適なハイパーパラメータを探索する．グリッドサーチを行うハイパーパラメータは，表 2 のように設定した．DBN の Fine-tuning には，ロジスティック回帰を用いた．

表 2: ハイパーパラメータ

DBN	隠れ層の構造	91, 137-91, 152-121-91 273, 273-273, 273-273-273[4]
	Pre-training の学習率	0.001, 0.01, 0.1
	Fine-tuning の学習率	0.001,0.01,0.1
	Pre-training の Epoch	500, 1000, 2000, 3000
	Fine-tuning の Epoch	500, 1000, 2000, 3000
MLP	隠れ層の構造	91, 137-91, 152-121-91 273, 273-273, 273-273-273
	学習率	0.001, 0.01, 0.1
	Epoch	500, 1000, 2000, 3000

[4]例えば 152-121-91 の場合，隠れ層が 3 層でユニット数が初期層から順に 152，121，91 であることを意味している．

4.2 実験結果及び考察

Validation error が小さい順に上位 5，10，15，20，25，30 の平均正解率[5]を，図 3，図 4 に示す（m：手動収集データ，a：自動収集データ，p：擬似データ）．

MLP と DBN の両方とも学習データに自動収集データと擬似データの両方を加えたほうが精度[6]が良くなる結果となった．MLP では，擬似データのみを加えた学習データは精度が良くなる場合があるのに対し，自動収集データのみを加えた学習データでは精度が悪くなった．逆に DBN では，擬似データのみを加えた学習データは精度が悪くなる場合が多くなるのに対し，自動収集データのみを加えた学習データは精度が良くなる場合が多い結果となった．これは，自動収集データは擬似データよりノイズを多く含むため，MLP はノイズで悪く影響を受けているが，DBN は Pre-training の段階でノイズで良い影響を受けていることを意味している．また，擬似データのみを加えた学習データでは，DBN の擬似データ 600 件を除けば，MLP と DBN であまり精度が変わらないことから，DBN の Pre-training を有効に活用できていないことが考えられる．MLP と DBN で共通して自動収集データのみを 2,400 件加えた学習データで著しく精度が悪くなっているのは，不正確なラベルが増えすぎたことが原因として考えられる．

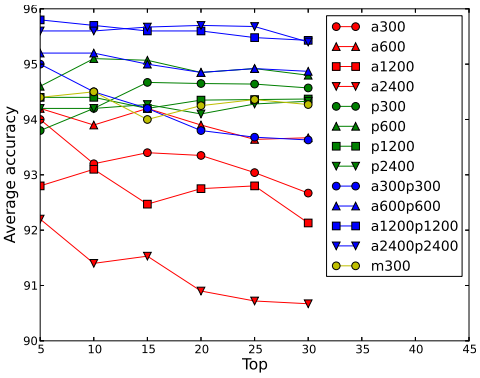

図 3: MLP の平均正解率

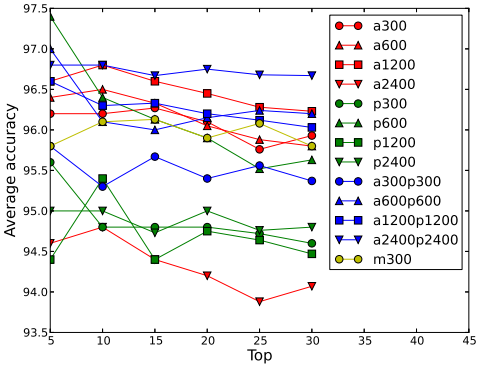

図 4: DBN の平均正解率

[5]評価データの数の少なさを補うために平均を取っている．
[6]評価データに対しての精度．

MLP と DBN を比較したものを図 5 に示す．

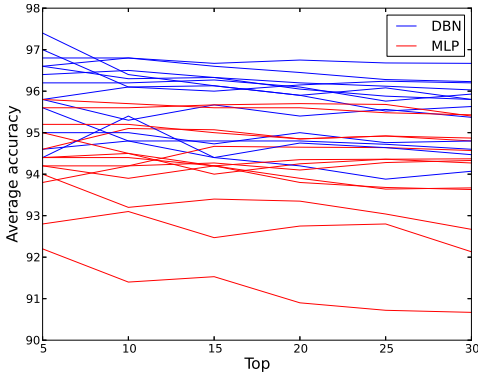

図 5: MLP と DBN の平均正解率の比較

図 5 を見てもわかるように，学習データによっては MLP の方が精度が良いものがあるものの，全体的に MLP より DBN の方が精度が良い結果となった．

すべての学習データにおける上位 30 までの隠れ層の構造の分布を表 3 に示す（隠れ層の構造の総数は 13 × 30 = 390）．

表 3: 隠れ層の構造の分布

隠れ層の構造	MLP の数	DBN の数
91	60	72
137 - 91	66	42
152 - 121 - 91	60	18
273	72	201
273 - 273	67	33
273 - 273 - 273	65	24

MLP の隠れ層の構造が平均正解数にあまり影響しないのに対し，DBN の隠れ層の構造は多層構造（137 - 91，152 - 121 - 91，273 - 273，273 - 273 - 273）より単層構造（91，273）のほうが多く，全体の 70 %を占める結果となった．これによって，一般には Deep Learning は多層構造のほうが良い結果となりやすいと言われている [3] ため予想とは異なる結果となってしまったが，言語処理の場合だと単層構造のほうが良い結果となる可能性が考えられる．また，隠れ層のユニットの数は多いほど，良くなる場合が多い [4] ことや，各層のユニットの数は同じにしたほうが良くなる場合が多い [5] ことが報告されているが，前者のほうは表 3 の隠れ層のユニット数の 273 のほうが 91 よりも精度が上位となる DBN の数が多いことからも裏付けられている．

5 おわりに

本稿では，関連語・周辺語からの検索用語の予測について DBN を用いた手法を提案し，DBN を用いたほうが MLP を用いるよりも精度がよいことを示した．また，自動収集データのようなノイズを多く含むようなものでも DBN では有効であり，自動収集データと擬似データを加えることで DBN だけでなく MLP でも有効であることを示した．

今回の実験結果からラベル（検索用語）の数が少ない場合，精度の高い検索用語の予測が可能であることがわかった．しかし，ラベルの数を増やすと精度の低下が考えられ，精度を向上させる方法としては，擬似データの欠損・ノイズを加える割合を変化させる，ベクトルを変換する際に様々な表記揺れにも対応できるようにする，関連語・周辺語としてラベルの前後の限られた範囲の名詞を用いる，最適なハイパーパラメータの探索にランダムサンプリングを用いる [4]，Drop connect 等の過学習抑制手法を取り入れる等が挙げられる．また，より精度の高い関連語・周辺語コーパスの自動収集の手法も確立する必要がある．

今後は上記の方策を取り入れながら大規模で高精度な検索用語予測システムの開発を行う予定である．

謝辞

本研究は科研費（25330368）の助成を受けたものである．

参考文献

[1] Yoshua Bengio: Learning Deep Architectures for AI, Foundations and Trends in Machine Learning, Vol. 2, No. 1, pp. 1-127, 2009

[2] Tijmen Tieleman: Training Restricted Boltzmann Machines using Approximations to the Likelihood Gradient, Proceedings of the 25th International Conference on Machine Learning, pp. 1064-1071, 2008

[3] Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton: ImageNet Classification with Deep Convolutional Neural Networks, `http://www.image-net.org/challenges/LSVRC/2012/supervision.pdf`, Imagenet Large Scale Visual Recognition Challenge, 2012

[4] Yoshua Bengio: Practical Recommendations for Gradient-based Training of Deep Architectures, `http://arxiv.org/abs/1206.5533v2`, 2012

[5] Pascal Vincent, Hugo Larochelle, and et al.: Stacked Denoising Autoencoders Learning Useful Representations in a Deep Network with a Local Denoising Criterion, The Journal of Machine Learning Research, Vol. 11, pp. 3371-3408, 2010