検索語の予測における Deep Learning と従来の機械学習との比較

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

昨今、Yahoo や Google といった企業が提供している検索エンジンは我々の生活に必要不可欠な存在となってきているが、現状では検索エンジンを使いこなすことが求められる場面が多々あると言わざるを得ない。例えば、適切な検索語が思い浮かばないことにより検索にかける時間が増えるといった問題がある。我々は関連語・周辺語からの検索語の予測・提示を行うシステム（例えば「計算機」、「頭脳」から「CPU」）の開発を目標にしている。その事前段階として検索語の分野を限定したシステムで Deep Learning の一種である Deep Belief Network（以降略して「DBN」と呼ぶ）を用いた手法を提案し、Multi Layer Perceptron（以降略して「MLP」と呼ぶ）との比較を行った [1]。さらに、Support Vector Machine（以降略して「SVM」と呼ぶ）との比較も行った [2]。しかしながら、教師付きデータは人手による収集のため実験データの規模が小さいことや正則化を行なっていないといった問題点が存在した。

本研究では教師付きデータを人手によるものから辞書サイトからの収集にすることにより収集コストを抑え、実験データの規模を拡大している。また、従来の正則化技術である L1 正則化、L2 正則化と Dropout と呼ばれるニューラルネットワーク正則化を加えることにより、どのような影響を与えるかを確認した。さらに、先行研究の比較対象である DBN、MLP、SVM に加え、Deep Learning のその他の手法である Stacked Denoising Autoencoders（以降略して「SdA」と呼ぶ）とベースライン手法として Bernoulli Naive Bayes（以降略して「BNB」と呼ぶ）を追加して比較を行った。

2 データの収集

関連語・周辺語から適切な検索語を予測する場合、それらの対応関係を表したデータが必要となる。適切な検索語を説明している文書には関連語・周辺語となる用語が多く含まれると考え、インターネットからの収集を行った。先行研究では誤りの少ないデータをコストの高い人手で収集したが、辞書サイトのデータを収集する方法に変更することで収集コストを抑えている。また、データの規模を考慮して対象となる検索語は収集対象の辞書サイト間でより共通して掲載されている 100 語としている。

2.1 辞書サイトデータ

検索語は IT・計算機関連の分野に限定しており、このような用語を説明する文書は複数のサイトに存在する。このようなサイトをここでは辞書サイトと呼ぶ。辞書サイトはサイトによって形式が異なるため、完全に自動で収集することは困難である。本研究では、辞書サイト毎にタグ情報を用いて検索語を説明している範囲を特定し抽出する正規表現スクリプトを作成することで、22 の辞書サイトを対象に収集を行った。この方法で収集された説明文書の集合を辞書サイトデータと呼ぶこととした。

2.2 検索エンジンデータ

辞書サイトデータより大規模なデータを収集する手段として、検索エンジンを利用することが考えられる。本研究では、説明文書に共通して出現する助詞に着目し、”検索語+助詞”をクエリとして Google 検索を行い、データ収集を行った。用いた助詞は「とは」・「は」・「というものは」・「については」・「の意味は」の 5 語である。このように収集された Web ページの集合を検索エンジンデータと呼ぶこととした。

2.3 擬似データ

検索エンジンデータのような適度にノイズを含むデータは大規模に収集できる利点がある反面、検索語と説明文書とされる Web ページの対応が不正確な場合がある。このような欠点を補完する考えとして、検索語との対応付けが正確なデータに意図的にノイズを加えることが挙げられる。本研究では、辞書サイトデータに対して 10%の割合で欠損、ノイズまたはその両方を加えたデータを擬似データと呼ぶこととした（詳細は [1] を参照されたい）。

3 Deep Learning 及び正則化

Deep Learning は教師なし学習器と教師あり学習器で構成されており、教師あり学習の事前に行われる教師なし学習の段階でより良い特徴が抽出されるように学習が行われる [3]。教師なし学習器は大別して確率的な動作を行う Restricted Boltzmann Machine（以降略し

て「RBM」と呼ぶ）と決定的な動作を行う Denoising Autoencoders（以降略して「dA」と呼ぶ）に分けられる。Deep Learning に RBM を用いたニューラルネットワークは DBN、dA を用いたニューラルネットワークは SdA と呼ばれる。

正則化は学習データに過剰に適合しないように制約を加えることである。L2 正則化は一般的に用いられる正則化であり、誤差関数に重みの 2 乗の総和を加えることで重みが非常に大きくならないように制限をかける。これに対して、L1 正則化は誤差関数に重みの絶対値の総和を加える正則化であり、重みの多くが 0 になるという特徴を有する。

Dropout はニューラルネットワークの一部のユニットを学習データ毎に無作為に P%取り除いて学習をすることで正則化を行う手法である [4]。推定時には全てのユニットの重みを（100 - P）%にすることで異なるモデルの幾何平均を取ることになる。

4 実験

4.1 実験条件

DBN、SdA の事前学習に用いる教師なし学習データとして辞書サイトデータ 1,134 件を基本に検索エンジンデータ 13 ,000 件、25,000 件、擬似データ 13,000 件、25,000 件、検索エンジンデータと擬似データを各 13,000 件、各 25,000 件を加えた場合に分けて、実験を行った。DBN と SdA の教師あり学習と他の学習器の学習には辞書サイトデータ 1,134 件、評価データとしては辞書サイトデータから取り除いた 300 件を用いた。DBN と SdA、MLP はバッチサイズが 32 のミニバッチ確率的勾配降下法で学習データに対する誤差（0/1 損失の平均）が許容誤差 0.03 以下になるまで教師あり学習を行なっている。グリッドサーチについては、各学習器のハイパーパラメータの組み合わせ（つまり、ハイパーパラメータセット）の数がほぼ同等になるように設定されており、その数は正則化パラメータを加えない実験で 243、正則化パラメータを加える実験で 2,187 となっている。また、グリッドサーチを行うハイパーパラメータとしての隠れ層は徐々にユニットを減らすような構造、同じ数のユニットを置いた構造、徐々に増やすような構造の 1 層から 3 層のものを、学習率は 0.05 から 0.5 の範囲のものを用いている。

辞書サイトデータ、検索エンジンデータ、擬似データの各データは文書であるため、ベクトルに変換する必要があるが出現する全ての単語をベクトルの要素とすると次元の呪いや計算時間の増加等が問題となる。ここでは先行研究 [1] のベクトル変換の手順に従い、ベクトルの要素を選定することで文書からベクトルへの変換を行なっており、ベクトルの次元の数は 1,323 となった。

4.2 実験結果

各ハイパーパラメータセットを検証誤差の小さい順に並べてその上位 N 個を用いたときの精度[1]の平均（ただし、N = 5, 10, 15, ⋯ , 100）を図 1、図 2、図 3 に示す。ただし、ds、se、ps はそれぞれ辞書サイトデータ、検索エンジンデータ、擬似データを表しており、その後ろに付いている数字はデータの数を表している。

図 1、図 2 から DBN、SdA ともに検索エンジンデータ、擬似データを事前学習データに含めて事前学習を行ったほうが結果が良くなっていることがわかる。効果の大小は検索エンジンデータ > 検索エンジンデータ + 擬似データ > 擬似データの順番となっており、擬似データのみでも効果があることがわかるが検索エンジンデータに擬似データを加えることで平均精度が下がる結果となってしまっている。

もっとも検証誤差の小さいハイパーパラメータセットの精度を表 1 に示す。

学習器	精度
DBN(ds1134)	0.780
DBN(ds1134, se25000)	0.793
SdA(ds1134)	0.770
SdA(ds1134, se25000)	**0.810**
MLP	0.780
SVM(Linear)	0.807
SVM(RBF)	0.717
BNB	0.773

表 1: ハイパーパラメータセット（N = 1）の精度

ただし、DBN、SdA については精度の高い場合、つまり、ds1134 と se25000 を用いた場合の結果のみを示している。もっとも精度が高かったのは SdA(ds1134, se25000) の 0.810 となっており、次いで SVM(Linear) の 0.807、DBN(ds1134, se25000) の 0.793 となっている。

図 3 に手法間の比較結果を示す。ただし、SVM(RBF) は精度が低いことから縦軸の範囲外（0.74 未満）となっているため表示されていない。図 3 を見てもわかるとおり、DBN(ds1134, se25000)、SdA(ds1134, se25000) は安定して他の従来の機械学習に比べて精度が高くなっている。これに対して、SVM(Linear) は右肩下がりの形で上位 20 くらいまでは DBN(ds1134, se25000) に近い精度となっているが、上位 100 までとなると SVM(RBF) に次いでの低い精度となっている。MLP の平均精度は DBN(ds1134)、SdA(ds1134) と近い精度となっているが、事前学習データを増やしたことにより DBN(ds1134, se25000)、SdA(ds1134, se25000) は差別化されていることが見て取れる。

MLP、DBN(ds1134, se25000)、SdA(ds1134, se25000) の精度と構造の関係を図 4、図 5、図 6 に示す。図の青の点は大きいほどその精度が多く現れていること

[1]評価データに対しての精度である。

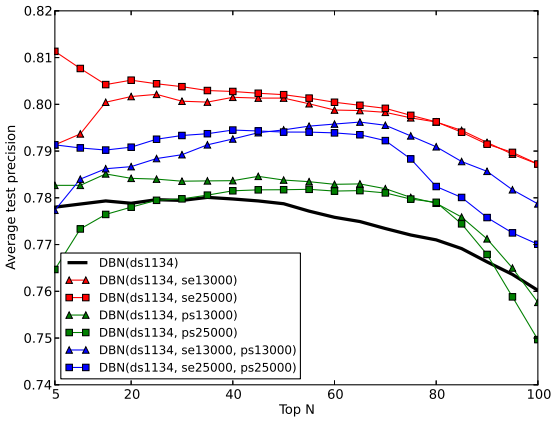

図 1: DBN の精度

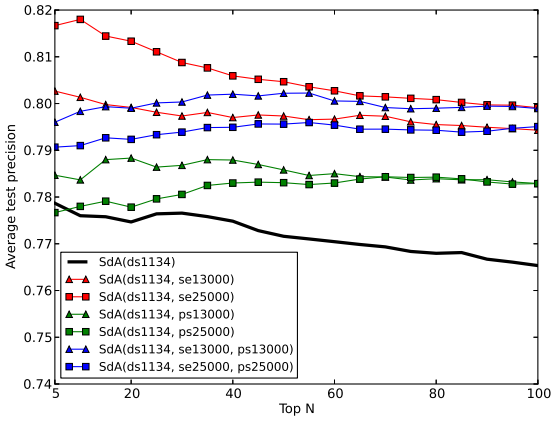

図 2: SdA の精度

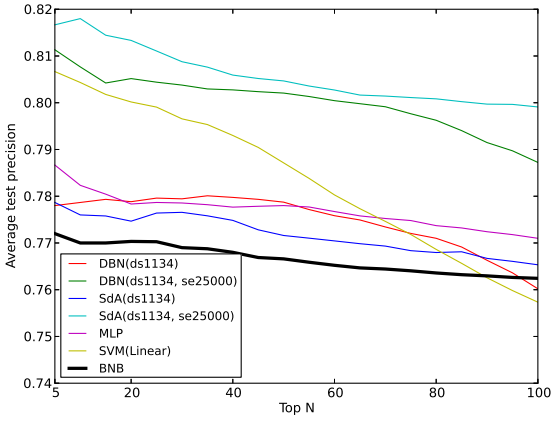

図 3: ベースラインと各機械学習手法の精度

を表している。図からは、MLP と SdA についてはその構造の違いによる精度の違いがあまり見られない。つまり、構造が単層でも多層でも青い丸はある程度均等に分布している。一方、MLP に比べて DBN は単層のほうが高い精度のところに青い丸が集中しており、単層か多層かによって精度に顕著な違いが見られる。SdA は事前学習に用いられる教師なし学習データが ds1134 と se25000 以外の場合でも、青い丸は均等な分布になっていることがほとんどであった。

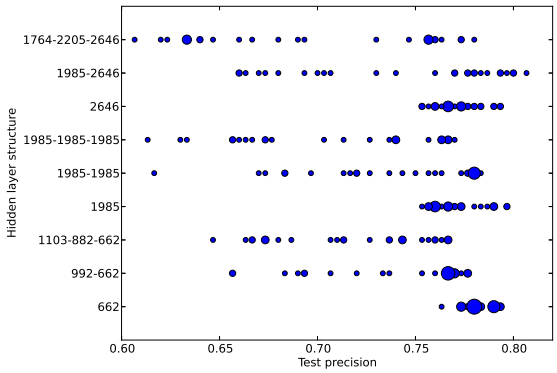

図 4: MLP の精度と構造

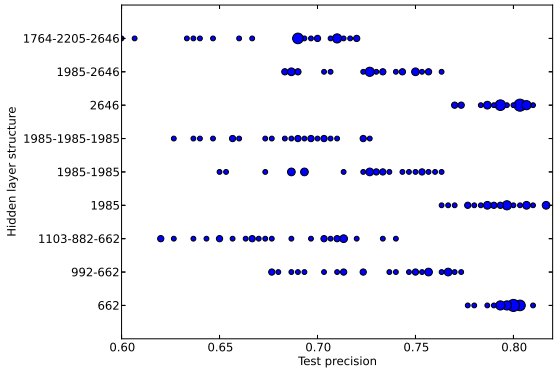

図 5: DBN の精度と構造

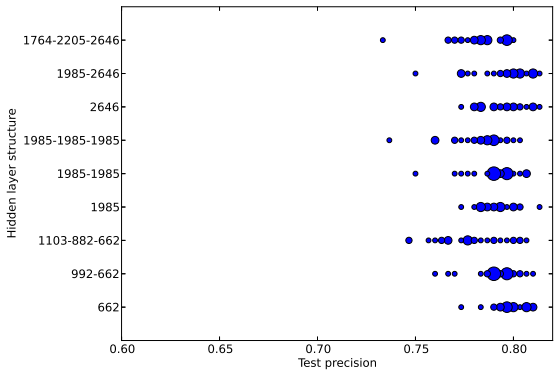

図 6: SdA の精度と構造

正則化パラメータ（L1 正則化係数、L2 正則化係数、入力層の Dropout の割合、隠れ層の Dropout の割合）をハイパーパラメータに加えた MLP、DBN(ds1134, se25000)、SdA(ds1134, se25000) の精度と検証誤差を図 7、図 8、図 9 に示す。実線は左軸の平均精度、破線は右軸の検証誤差に対応している。正則化パラメータを加えると、ハイパーパラメータの組み合わせの数が 2187 と非常に多くなるため、事前学習に全学習データセットを用いると、グリッドサーチに膨大な時間がかかってしまう。そのため、事前学習にはもっとも精度の高かった学習データセット（ds1134、se25000）のみを用いることにした。全体的に L1 正則化と L2 正則化は効果があまり見られなかったのに対し、Dropout

は学習器に関わらず検証誤差が低下している。これに対して、精度が向上していない理由としては様々なことが考えられるが評価データの規模が小さいことによる偏りやグリッドサーチの探索範囲の拡大に伴うハイパーパラメータの最適化が不十分、学習データの検索語の数の偏り等の理由が考えられる。また、MLP は L1 正則化、L2 正則化を行うことで多少精度が向上しているが、Deep Learning の精度よりは低い結果となっている。

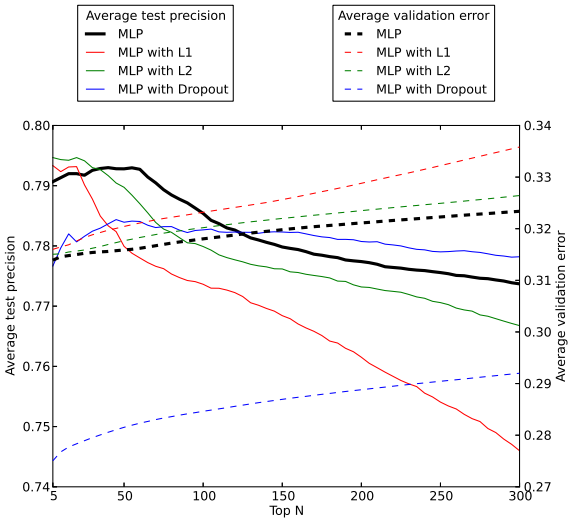

図 7: MLP の平均精度と検証誤差（正則化）

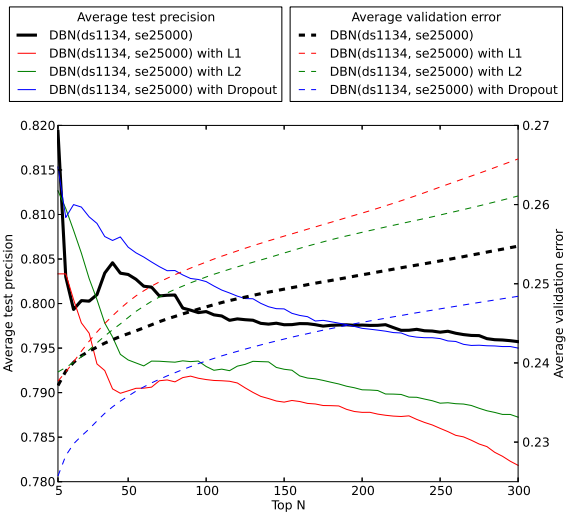

図 8: DBN の平均精度と検証誤差（正則化）

5 おわりに

本稿では、検索語の規模を大きくした場合にも、関連語・周辺語からの検索語の予測は、Deep Learning のほうが従来の機械学習手法より精度が高いことを示した。また、教師付きデータには人手による収集でなくとも辞書サイトのデータを用いても有効であることや検索エンジンデータや擬似データを事前学習に加えることで検索語の規模が変化しても精度が向上することを示した。今後としては更に検索語の規模を拡大した実験では精度にどのような影響を与えるか、検索語の規模を拡大しても精度が低下しない手法の考案、説明文書以外を対象としたデータの収集・構築が課題として考えられる。

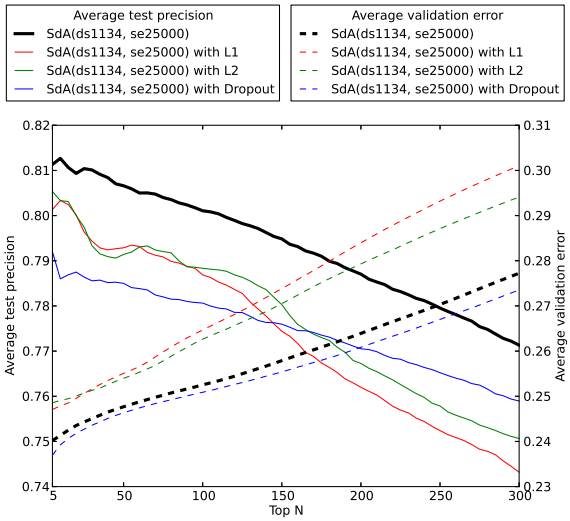

図 9: SdA の平均精度と検証誤差（正則化）

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] 谷河, 馬, 村田. 2014. Deep Belief Network を用いた関連語・周辺語からの検索用語の予測. 言語処理学会第 20 回年次大会, 547-550.

[2] Q. Ma, I. Tanigawa, and M. Murata. 2014. Retrieval Term Prediction Using Deep Belief Networks. *The 28th Pacific Asia Conference on Language, Information and Computing*.

[3] Y. Bengio. 2009. Learning Deep Architectures for AI. *Foundations and Trends in Machine Learning*, 2(1):1-127.

[4] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov. 2012. *Improving neural networks by preventing co-adaptation of feature detectors*, arXiv:1207.0580.