

文字レベル深層学習の日本語データセットへの応用

佐藤 拳斗^{1,a)} 折原 良平^{1,b)} 清 雄一^{1,c)} 田原 康之^{1,d)} 大須賀 昭彦^{1,e)}

概要：近年、畳み込みニューラルネットワーク (ConvNets) をテキスト分類に適用するにあたり、表音文字レベルでの畳み込みを行うことにより、形態素解析等の言語に依存する処理を必要としない手法が提案されている。先行研究においてはニューステキストのカテゴリ分類やレビューデータの極性分析を題材として、英語と中国語のデータセットにおいてその有効性が検証されてきた。本研究では日本語のデータセットにおいてその有効性を検証したため、その結果を報告する。また、ConvNet が中間層において抽出した特徴量について N-gram 特徴量と比較、分析した結果も合わせて報告する。

1. はじめに

近年、メディア情報処理の分野において、深層学習 (Deep Learning) の研究が盛んになってきている。中でも、物体認識、画像認識の分野においては、畳み込みニューラルネットワーク (Convolutional Networks; CNNs, ConvNets) を深層化することにより飛躍的に精度が向上した。自然言語処理の分野における例としては、word2vec [14], [15], [16] と呼ばれる単語の概念ベクトルを分散表現 (Distributed Representations) として獲得する手法が有名である。自然言語処理の中でも、深層学習をテキスト分類や感情分析に用いる研究が盛んに行われている。

従来のテキスト分類、感情分析に関する研究では、いかにして分類に有用な特徴量を作成するかが重要であった。例えば、分類したい言語についての形態素解析器をあらかじめ作成しておき、形態素解析の結果に基づきトークン化し、Bag-of-Words (BoW) モデルや Bag-of-N-grams モデルを作成し、サポートベクトルマシン (SVM) やロジスティック回帰等の機械学習手法を組み合わせるといったものである。

深層学習を用いたテキスト分類、感情分析に関する研究として、Glorot ら [8] の Amazon レビューデータに対し積層型雑音除去自己符号化器 (Stacked Denoising Autoencoders; SDAs) により感情分析を行うという研究がある。こちらの研究では入力として BoW を用いている。その他には、

Socher ら [22] の構文木に基づく再帰ニューラルネットワーク (Recursive Neural Networks; RNNs) により感情の分布を予測するという研究がある。

以上の研究では、いずれも形態素解析器や構文解析器、特徴語辞書等、言語に依存するシステムが必要となり、応用先が別の言語となった場合に、それらをまた一から用意する必要があるといったデメリットがある。

一方で、1次元畳み込みニューラルネットワーク (Temporal Convolutional Neural Networks; Temporal CNNs, ConvNets) をテキスト分類に適用する事例が報告されている [5], [6], [10], [19], [20], [25], [26]。1次元 ConvNet 用いた手法の中には、単語レベルで ConvNet を適用する手法と、文字レベル、特に表音文字レベルで ConvNet を適用する手法、両者を組み合わせた手法とがあり、文字レベル ConvNet は形態素解析等の言語に依存したシステムが不要となる。先行研究において、文字レベル ConvNet は英語のみならず中国語のデータセットにおいても有効である事が確認された。

本研究では、日本語のデータセットにおいて、表音文字レベル ConvNet を適用し、その有効性を検証したためその結果を報告する。また、ConvNet が中間層において抽出した特徴量について未だ議論がなされていなかったため、N-gram 特徴量と比較、分析した結果も合わせて報告する。こちらは、日本語のデータセットで学習した結果と英語のデータセットで学習した結果両方について調査した。

2章では ConvNet をテキスト分類に適用した関連研究を、3章では本研究で用いる文字レベル ConvNet のモデルを、4章では実験とその結果を説明し、5章ではディスカッションを行い、最後に6章ではまとめと今後の課題について述べる。

¹ 電気通信大学大学院情報システム学研究科
Graduate School of Information Systems, University of
Electro-Communications, Chofu-shi, Tokyo 182-8585, Japan
a) sato.minato@ohsuga.is.uec.ac.jp
b) ryohei.oriyama@toshiba.co.jp
c) sei@is.uec.ac.jp
d) tahara@is.uec.ac.jp
e) ohsuga@uec.ac.jp

2. 関連研究

2.1 ConvNet をテキスト分類に適用した研究

2.1.1 単語レベル ConvNet に関する研究

単語レベルの ConvNet をテキスト分類,感情分析に適用した研究として, Kim[10] の研究や, Severyn ら [19], [20] の研究がある. これらの研究は, テキストを word2vec 等によって事前学習した単語の分散表現の系列として, 1 次元畳み込み (Temporal Convolution), 1 次元最大プーリング (Temporal Max-Pooling) を行うというものである. Severyn らの研究では, Twitter や SMS といった比較的短めのテキストの感情分析においての有効性を検証した. Kim の研究では, 複数の Window Size の畳み込みフィルタを用いる事によって, 映画の評判分析について良い結果が得られている.

2.1.2 文字レベル ConvNet に関する研究

表音文字レベルの ConvNet をテキスト分類,感情分析に応用した研究として, Santos ら [5], [6] の研究や, Zhang ら [25], [26] の研究がある. Santos らの研究では, ランダムに初期化された文字の分散表現の系列として, 1 次元方向に畳み込み, プーリングを行い, そこで得られた特徴量と単語レベルの畳み込みを行った時に得られる特徴量を組み合わせる事によって精度向上を図った. Zhang らの研究では, 文字を離散表現 (Discrete Representations) の一つである one-hot 表現と呼ばれる, 対応する特定の文字に対応する次元だけ 1, 他の要素が全て 0 というベクトルで表した. テキストをそれらの one-hot 表現の系列として扱い, 1 次元畳み込み, 1 次元最大プーリングの処理を行うのだが, その際ネットワークを深層化することにより精度向上を図った. その結果, 英語のデータセットだけでなく, 中国語のデータセットにおいても有効性を示した. その際, 中国語は文字の種類が多く, one-hot 表現の次元が莫大になってしまうので, pypinyin という Python のライブラリを用いてローマ字化を行い, one-hot 表現の次元を英語と同等にした.

3. 文字レベル ConvNet

3.1 概要

3.1.1 入力表現

N 文字からなる文 ($\{c_1, c_2, \dots, c_N\}$) が与えられた場合を考える. それぞれの文字 c_n は文字種類数 d 次元の one-hot 表現 $r_n \in \mathbb{R}^d$ に変換され, N 文字からなる文は $\{r_1, r_2, \dots, r_N\} \in \mathbb{R}^{d \times N}$ となる.

3.1.2 畳み込み

Window Size が k である畳み込みフィルタの適用範囲 $z_n \in \mathbb{R}^{d \times k}$ を考えると,

$$z_n = (r_{n-(k-1)/2}, \dots, r_{n+(k-1)/2})^T \quad (1)$$

となる. これらから, i 番目の畳み込みフィルタによる z_n に対する畳み込みの出力は,

$$[u_n]_i = [Wz_n + b]_i \quad (2)$$

ここで, $W \in \mathbb{R}^{f \times d \times k}$, $b \in \mathbb{R}^f$ は, それぞれ f 枚の畳み込みフィルタから構成される重み, バイアスを表しており, 誤差逆伝播法 (Error Backpropagation) によって学習するパラメータである. 畳み込み層において, パディングと呼ばれる仮想的な 0 ベクトルの系列を入力の前後に設けなかった場合は, 畳み込み層の出力の次元は $f \times (N - (k - 1))$ となる.

3.1.3 プーリング

$M = (N - (k - 1))$ としたとき, 畳み込み層における i 番目の畳み込みフィルタの出力は, $(v_1, v_2, \dots, v_M)_i \in \mathbb{R}^{N-(k-1)}$ と表すことができる. 最大プーリングサイズ p を用いて, プーリングが適用される範囲 $[y_m]_i$ は,

$$[y_m]_i = (v_{m-(p-1)/2}, \dots, v_{m+(p-1)/2})_i \quad (3)$$

となり, プーリング層の出力の次元は, $f \times (N - (k - 1)) / p$ となる.

3.2 モデルの設計

3.2.1 Zhang ら [25], [26] のモデル

Zhang ら [25], [26] は, 6 層の畳み込み層と 3 層の全結合層のモデルを構築した. そのモデルには, 畳み込みフィルタの枚数の多い Large と少ない Small の 2 つがある. 畳み込み層のパラメータについては表 1 に, 全結合層のパラメータについては表 2 に示した. 表 1 の列はそれぞれ, 層の番号, Large の畳み込みフィルタの枚数, Small の畳み込みフィルタの枚数, 畳み込みフィルタの Window Size, 最大プーリングのサイズを表している. 表 2 の列はそれぞれ, 層の番号, Large の各層の素子数, Small の各層の素子数を表している. 第 9 層は出力層となっており, 扱う問題によって素子数が変わってくる (e.g., ポジティブ/ネガティブの 2 値を判定する場合は素子数は 2 となる.). このモデルを以下 Zhang(Large), Zhang(Small) とする.

表 1 Zhang ら [25], [26] のモデル (畳み込み層)

Layer	Large Frame	Small Frame	Window	Pool
1	1024	256	7	3
2	1024	256	7	3
3	1024	256	3	N/A
4	1024	256	3	N/A
5	1024	256	3	N/A
6	1024	256	3	3

表 2 Zhang ら [25], [26] のモデル (全結合層)

Layer	Output Units Large	Output Units Small
7	2048	1024
8	2048	1024
9	Depends on the Problem	

本研究における入力の文字は次の 68 文字である .

```
abcdefghijklmnopqrstuvwxyz0123456789
,;.:!?:'"/\|_@#%&*'~'"+-=<>(){}
```

また, 入力の文の長さは 1014 となっている . そのため, 第 6 層目の出力の長さは $(1014 - 96) / 27 = 34$ となり, 同層の出力の素子数 (*i.e.*, 第 7 層目の入力の素子数) は Large モデルの場合 $34 \times 1024 = 34816$ となる .

出力層を除くいずれの層の活性化関数にも ReLU[17] を用いた . 誤差逆伝播法により学習を行う際の最適化アルゴリズムにはモーメント付き確率的勾配降下法 (Momentum SGD)[1] を用いた . その際, ミニバッチサイズ 50, モーメント項の係数は 0.9 で学習率は初期値 0.01, 3 epoch 毎に半減させ合計 30 epoch 学習させた . また, 正則化手法として 3 層ある全結合層の間に Dropout[23] を設けている . いずれも Dropout の比率は 0.5 となっている . ネットワークの重みの初期化については, 先行研究では, Large モデルの場合, 平均 0 / 標準偏差 0.02, Small モデルの場合, 平均 0 / 標準偏差 0.05 となるガウス分布に基づく乱数で行っていたが, Glorot ら [7] の一様乱数に基づく初期化を用いた .

3.2.2 本研究で追加したモデル

本研究において, 表 3, 表 4 に示される, よりシンプルな畳み込み層, プーリング層を各 1 層ずつ設けたモデル (以下 Conv1(Large), Conv1(Small)) を用意した . これは, 本研究で使用する日本語のデータセットの規模が比較的小さく, それに対して Zhang らのモデルの表現力が強すぎるため, テキスト分類に対する ConvNet の有効性を検証するのに難しいと考え, 表現力の弱いモデルを用意した .

表 3 Conv1(畳み込み層)

Layer	Large Frame	Small Frame	Window	Pool
1	1024	256	7	1008

表 4 Conv1(全結合層)

Layer	Output Units Large	Output Units Small
2	Depends on the Problem	

こちらのモデルも, 基本的には 3.2.1 項で述べた学習アルゴリズムを用いたが, Dropout は行わなかった .

4. 実験

4.1 比較手法

比較手法として Bag-of-Words モデルと Bag-of-N-grams モデルを用意した .

Bag-of-Words 英語のデータセットにおいては, Python のライブラリである gensim[18] を用いて前処理等を行い, ストップワードを除く最頻出の 5000 語を辞書として用いた . 日本語のデータセットにおいては, 日本語形態素解析器である MeCab[11] を用いて形態素解析を行い, 最頻出の 5000 語を辞書として用いた .

Bag-of-N-grams 英語のデータセットにおいては, 1 から 5gram のうち最頻出の 5000 語を辞書として用いた . 日本語のデータセットにおいては, 逆かな (ローマ字) 漢字変換ツールである KAKASI^{*1} を用いて, ローマ字に変換した後に英語のデータセットと同様の処理で辞書を作成した .

Bag-of-Words モデルと Bag-of-N-grams モデル両方について, 辞書を基に tf-idf 特徴量を作り, 多項ロジスティック回帰を用いて学習, 分類を行った .

4.2 ConvNet の実装

実装面に関しては, Python のライブラリである Keras[2] を用いて実装を行った . Keras のバックエンドエンジンには同ライブラリの Theano[24] を用いた .

4.3 データセットと結果

4.3.1 AFPBB データセット

日本語のデータセットとして, フランス通信社 AFP の日本語版^{*2}の記事を用いた . 2006 年 5 月から 2016 年 5 月の記事を独自に収集した . データセットの概要を表 5 に示す . “Lifestyle”, “Politics”, “Science”, “Sports” の 4 カテゴリーがあり, それぞれ訓練用に 12,000 件, 検証用に 500 件, 評価用に 500 件サンプリングした . また, Bag-of-N-grams モデルと ConvNet の入力として, 日本語文字列を KAKASI でローマ字化したものを用いた . 下記は, その 1 例である .

ポーランドで 21 日に行われた総選挙は, 出口調査の …

↓ KAKASI によるローマ字化

```
po^randode21nichiniokonawaretasousenkyoha,
deguchichousano...
```

一部例からもわかる通り, 長音符 (ー) はハット (^) に, 句点 (。) 読点 (、) はそれぞれピリオド (.), カンマ (,), 中黒 (・) はピリオド (.) に置き換えられる .

結果を表 6 に示す . 分類精度が最も良かったモデルは

*1 <http://kakasi.namazu.org>

*2 <http://www.afpbb.com>

表 5 AFPBB データセット

Category	Total	Train	Validation	Test
Lifestyle	21,927	12,000	500	500
Politics	18,221	12,000	500	500
Science	13,069	12,000	500	500
Sports	26,561	12,000	500	500

Bag-of-Words であったが、これは日本語の同音異義語を区別することができる等、MeCab による形態素解析の恩恵が寄与していると考えられる。その他のモデルの入力は KAKASI でローマ字化してあるため、同音異義語の区別がつかなくなっている等重要な情報が削減されてしまっている可能性がある。次点は Conv1(Small) となったが、これは畳み込みフィルタにより N-gram 特徴量よりも効果的な特徴量を得ることができているからと考えられる。また、Conv1 が Zhang らのモデルよりも良い結果になったのは、訓練用データセットのサンプル数が各クラス 12,000 件、合計 48,000 件と他のデータセットより比較的少なく、表現力の強い Zhang らのモデルが過学習を起こしてしまっているからと考えられる。

表 6 AFPBB データセットに対する結果

Model	Accuracy
Bag-of-N-grams	0.9335
Bag-of-Words	0.958
Conv1(Small)	0.943
Conv1(Large)	0.9405
Zhang(Small)	0.926
Zhang(Large)	0.9165

4.3.2 AG ニュースデータセット

英語のデータセットとして AG's corpus of news articles (以下 AG ニュースデータセット) [3], [9] を Gulli らのウェブサイト *3 から入手した。記事を多く含む “World”, “Sports”, “Business”, “Sci/Tech” の 4 カテゴリから表 7 に示す通りにサンプリングした。

表 7 AG ニュースデータセット

Category	Total	Train	Validation	Test
World	186,674	90,000	5,000	5,000
Sports	118,103	90,000	5,000	5,000
Business	134,223	90,000	5,000	5,000
Sci/Tech	153,595	90,000	5,000	5,000

結果を表 8 に示す。分類精度がもっとも良かったモデルは Conv1(Large) となったが、その他のモデルとさほど差はなかった。カテゴリ分類のタスクは、入力中に含まれる単語が結果につながりやすい。例えば、スポーツ選手の名前等はスポーツの記事に出現しやすく、且つ政治や科学等の記事には出現しにくいといったことが挙げられる。その

*3 https://www.di.unipi.it/~gulli/AG_corpus_of_news_articles.html

ため、今回のデータセットにおいては複雑な系列情報を考慮する必要性がさほどなく、単語を特徴として扱えばある水準の精度が保証され、Conv1 や Zhang らのモデル、Bag-of-Words モデル、Bag-of-N-grams モデルの結果にさほど差がなかったと推察される。

表 8 AG ニュースデータセットに対する結果

Model	Accuracy
Bag-of-Words	0.8689
Bag-of-N-grams	0.87665
Conv1(Small)	0.8701
Conv1(Large)	0.8849
Zhang(Small)	0.87095
Zhang(Large)	0.87925

4.3.3 Amazon レビューデータセット

英語のデータセットとして Amazon レビューデータセット [12], [13] を Stanford Network Analysis Project (SNAP) のウェブサイト *4 から入手した。“Books”, “Electronics”, “Movies and TV”, “CDs and Vinyl”, “Clothing, Shoes and Jewelry”, “Home and Kitchen”, “Kindle Store”, “Sports and Outdoors” の合計 8 カテゴリからレビューの評価の数が 5 つと 1 つのデータを表 9 に示す通りにサンプリングし、それぞれのデータをポジティブ、ネガティブとして極性分類を行った。なお、Bag-of-Words モデルと Bag-of-N-grams モデルにおける辞書は、各クラス 100,000 件ずつサンプリングして作成した。

表 9 Amazon レビューデータセット

Polarity	Total	Train	Validation	Test
Positive(5)	8,829,533	300,000	40,000	40,000
Negative(1)	653,333	300,000	40,000	40,000

結果を表 10 に示す。分類精度がもっとも良かったモデルは Zhang(Large) となった。ConvNet 系のモデルにおいて Bag-of-Words モデルや Bag-of-N-grams モデルと比べて、約 3~5%の精度向上を確認した。これは、カテゴリ分類タスクと比べて感情分析タスクでは複雑な系列情報を考慮する必要性があることに起因するからと考えられる。

表 10 Amazon レビューデータセットに対する結果

Model	Accuracy
Bag-of-Words	0.882175
Bag-of-N-grams	0.881275
Conv1(Small)	0.91125
Conv1(Large)	0.925425
Zhang(Small)	0.92155
Zhang(Large)	0.931925

*4 <https://snap.stanford.edu/data/web-Amazon.html>

5. ディスカッション

5.1 畳み込みフィルタの重みについて

図 1 は, Amazon レビューデータセットに対し Zhang(Small) で学習した第 1 層目のある畳み込みフィルタを可視化したものである. 横軸は文字の id, 縦軸は時系列方向 (つまり, Window Size 分だけある) を表しており, それぞれ白が大きい負の値, グレーが 0 に近い値, 黒が正の大きい値を表している. 図の上に対応する文字を表示してあるが, これからもわかる通りアルファベット abc ... xyz に対応する部分の分散が大きいことがわかる. これは, その他の記号よりもアルファベットについて特徴を学習していることを意味している. 図 2 は, 同学習後の畳み込みフィルタをランダムに選んで表示させたものである. 先の現象は図 2 からわかる様に, 他の畳み込みフィルタでも起こっていることがわかる. AG ニュースデータセットにおいても同様の現象を確認した. このことは Zhang らの先行研究の論文において言及されていた.

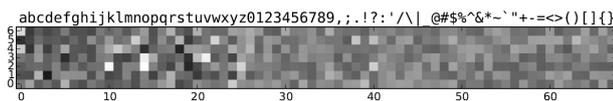


図 1 畳み込みフィルタの重みと対応する文字の可視化

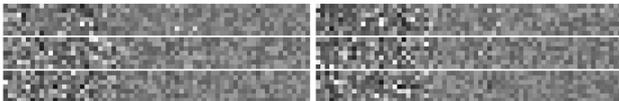


図 2 畳み込みフィルタの重みの可視化 (Amazon レビューデータ)

一方, AFPBB データセットに対し同様に学習した第 1 層目のある畳み込みフィルタが図 3 である. こちらはアルファベットとその他の記号とではさほど変わらない結果となったが, これは, データセットが他のものと比べてサンプル数が比較的少なく, 第 1 層目の畳み込みフィルタにおいてより汎用的な特徴を抽出するに至っていないためだと考えられる.

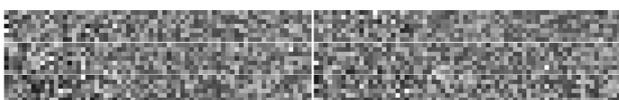


図 3 畳み込みフィルタの重みの可視化 (AFPBB データセット)

5.2 畳み込みフィルタに強く反応する N-gram について

5.1 節の畳み込みフィルタの重みの可視化は, 先行研究においてもなされていたが, 重みの可視化だけでは具体的にはどういった特徴を抽出しているか理解困難である. そこで, 畳み込みフィルタに強く反応する (*i.e.*, 畳み込みフィルタとの畳み込み演算の出力値が大きい) N-gram 特徴量について調査した. 本研究で用いた ConvNet のモデルの第 1 層目の畳み込みフィルタの Window Size は 7 であったので, 評価用のデータセットから 7-gram を生成し, ある畳み込みフィルタとの畳み込み演算を行い, その出力値のランキングを作った. 表 11 は AFPBB データ

表 11 ある畳み込みフィルタによる畳み込み演算の出力値ランキング (バイアス項を除く). AFPBB データセットに対して Conv1(Large) で学習した結果を用いて評価用データセットで計算した.

7-gram	Convolution Ouptut Value
,1-2deg	0.50774
a3-2deg	0.50613
a1-1deh	0.49199
a2-2deg	0.48423
a3-2deb	0.48332
a3-2def	0.47848
a1-3deh	0.47644
a3-1deh	0.47461
a2-1deg	0.47368
,3-2deh	0.46938

セットに対して Conv1(Large) で学習を行った結果である. これらの文字列にマッチングしている記事の部分文字列を調べた結果, “~は数字 (1 桁)-数字 (1 桁) で逆転 (あるいは引き分け)” となっているものが多々見受けられ, これはスポーツの記事に多く含まれていた. 実際, 表 11 の N-gram の文字列を先頭から全て OR で連結した正規表現 [, a][132][\-][213][d][e][ghbf] に一致する文字列を含む記事数をカテゴリごとに集計した結果, “Sports” が 20 件, その他カテゴリは全て 0 件となり畳み込みフィルタの特徴抽出が働いていることがわかる. つまり, 少なく見積もっても約 $2 \times 3 \times 1 \times 3 \times 1 \times 1 \times 4 = 72$ 通りの 7-gram をほぼ単一の特徴量として 1 枚の畳み込みフィルタで表現できていることになる. また, 仮に “逆転” や “gyakuten” を特徴量とするとしても, 選挙の逆転当選や支持率の逆転を報道する記事があった場合, この特徴量のみから “Sports” であると判断することが誤りあり, 畳み込みフィルタで学習された特徴量の優位性が見て取れる.

英語のデータセットである AG ニュースデータセットについても同様の処理を行った. 表 12 は AG ニュースデータセットに対して Zhang(Small) で学習を行った結果である. これらの文字列にマッチングしている記事の部分文字列を調べた結果, “save (or above or raise) \$数字”

表 12 ある畳み込みフィルタによる畳み込み演算の出力値ランキング (バイアス項を除く) . AG ニュースデータセットに対して Zhang(Small) で学習した結果を用いて評価用データセットで計算した .

7-gram	Convolution Oupput Value
ave \$70	1.02668
ove \$70	0.98627
ave \$10	0.97948
ts: \$10	0.96571
ise \$10	0.95659
the \$19	0.95559
9;s \$10	0.94505
ove \$10	0.93909
ave \$72	0.93072
ave \$20	0.93069

となっており、企業の経費削減や原油の価格高騰等のビジネスに関するものが大半を占めていた . 先と同様に、表 12 の N-gram の文字列を先頭から全て OR で連結した正規表現 [aoti9][vsh;][e:s][][\\$] [712] [092] に一致する文字列を含む記事数をカテゴリごとに集計した結果、“Business” が 22 件、“Sports” が 0 件、“World” が 1 件、“Sci/Tech” が 3 件となり、畳み込みフィルタによる特徴抽出が働いていることがわかる .

6. まとめ

本研究では、文字レベル ConvNet を日本語のデータセットに適用し、その有効性を検証した . また、中間層において得られる特徴量について、N-gram 特徴量と比較することで 1 枚の畳み込みフィルタで効果的な特徴を抽出することができていることを確認した .

今回用いた日本語のデータセットのサンプル数は数万オーダーと比較的小さいものであったため、ConvNet を深層化した際の効果の検証が難しかった . また、Bag-of-Words が最も良い精度となったが、MeCab による形態素解析が結果に大きく貢献していることが考えられるので、今後の課題としては、形態素解析の難しいくだけた表現の多い日本語のデータセット (e.g., Twitter) や、より大きな日本語データセットにおける検証が挙げられる .

また、画像認識分野においては、ImageNet[4] と呼ばれる大規模教師付き画像データセットで学習した ConvNet を用いて、別のデータセットの特徴を抽出し、線形 SVM で分類を学習するといった研究 [21] 等がある . これらの研究を参考に、大規模データセットを用いて学習された文字レベル ConvNet のネットワークの重みを用いた特徴抽出、転移学習を検討したい .

謝辞 本研究は JSPS 科研費 26330081, 26870201, 16K12411 の助成を受けたものです .
 本研究を遂行するにあたり、研究の機会と議論・研鑽の

場を提供して頂き、御指導頂いた国立情報学研究所 / 東京大学 本位田 真一 教授をはじめ、活発な議論と貴重な御意見を頂いた研究グループの皆様感謝致します .

参考文献

- [1] Yoshua Bengio, Nicolas Boulanger-Lewandowski, and Razvan Pascanu. Advances in Optimizing Recurrent Networks. In *Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP '13*, 2013.
- [2] François Chollet. Keras, 2015.
- [3] Gianna M. Del Corso, Antonio Gulli, and Francesco Romani. Ranking a Stream of News. In *Proceedings of the 14th International Conference on World Wide Web, WWW '05*, pp. 97–106, 2005.
- [4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, CVPR '09*, 2009.
- [5] Cícero Nogueira dos Santos, Bing Xiang, and Bowen Zhou. Classifying Relations by Ranking with Convolutional Neural Networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics, ACL '15*, pp. 626–634, 2015.
- [6] Cicero dos Santos and Maira Gatti. Deep Convolutional Neural Networks for Sentiment Analysis of Short Texts. In *Proceedings of the 25th International Conference on Computational Linguistics, COLING '14*, pp. 69–78, 2014.
- [7] Xavier Glorot and Yoshua Bengio. Understanding the Difficulty of Training Deep Feedforward Neural Networks. In *Proceedings of the 13rd International Conference on Artificial Intelligence and Statistics, AISTATS '10*, 2010.
- [8] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Domain Adaptation for Large-scale Sentiment Classification: A Deep Learning Approach. In *Proceedings of the 28th International Conference on Machine Learning, ICML '11*, 2011.
- [9] Antonio Gulli. The Anatomy of a News Search Engine. In *International Conference on World Wide Web, special interest tracks and posters, WWW '05*, pp. 880–881, 2005.
- [10] Yoon Kim. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP '14*, pp. 1746–1751, 2014.
- [11] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying Conditional Random Fields to Japanese Morphological Analysis. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, EMNLP '04*, pp. 230–237, 2004.
- [12] Julian McAuley, Rahul Pandey, and Jure Leskovec. Inferring Networks of Substitutable and Complementary Products. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '13*, pp. 785–794, 2015.
- [13] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. Image-Based Recommendations on Styles and Substitutes. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval,*

- SIGIR '13, pp. 43–52, 2015.
- [14] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. *ICLR '13*, 2013.
 - [15] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of the 27th Annual Conference on Neural Information Processing Systems*, NIPS '13, pp. 3111–3119. 2013.
 - [16] Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic Regularities in Continuous Space Word Representations. In *Proceedings of the 2013 Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL-HLT '13, pp. 746–751, 2013.
 - [17] Vinod Nair and Geoffrey E. Hinton. Rectified Linear Units Improve Restricted Boltzmann Machines. In *Proceedings of the 27th International Conference on Machine Learning*, ICML '10, pp. 807–814, 2010.
 - [18] Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pp. 45–50, 2010.
 - [19] Aliaksei Severyn and Alessandro Moschitti. Twitter Sentiment Analysis with Deep Convolutional Neural Networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, pp. 959–962, 2015.
 - [20] Aliaksei Severyn and Alessandro Moschitti. UNITN: Training Deep Convolutional Neural Network for Twitter Sentiment Classification. In *Proceedings of the 9th International Workshop on Semantic Evaluation*, SemEval '15, pp. 464–469, 2015.
 - [21] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. CNN Features Off-the-Shelf: An Astounding Baseline for Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, workshops*, CVPR '14, 2014.
 - [22] Richard Socher, Jeffrey Pennington, Eric H. Huang, Andrew Y. Ng, and Christopher D. Manning. Semi-Supervised Recursive Autoencoders for Predicting Sentiment Distributions. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, 2011.
 - [23] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, Vol. 15, pp. 1929–1958, 2014.
 - [24] Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, Vol. abs/1605.02688, , May 2016.
 - [25] Xiang Zhang and Yann LeCun. Text Understanding from Scratch. *CoRR*, Vol. abs/1502.01710, , 2015.
 - [26] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level Convolutional Networks for Text Classification. In *Proceedings of the 29th Annual Conference on Neural Information Processing Systems*, NIPS '15, pp. 649–657, 2015.