3D2-OS-37b-4

# Deep Learningを用いた人工言語の学習と獲得について

## Learning and Acquisition of Artificial Language using Deep Learning

岡田龍治[*1] 河原崎徳之[*1]
Ryuji Okada Noriyuki Kawarazaki

[*1] 神奈川工科大学
Kanagawa Institute of Technology

In this paper, we propose the optimum acquisition method of the artificial language using deep learning. At first, we apply the 1-of-k encoding process to the C language programs written by the students in our university in order to create the training data. Then, the training data are classified in characters and strings based on the 1-D convolution. Further, we use the RNN to obtain the sequence of the character in the strings and that of the string in the sentence. We clarified the effectiveness of our training model using 1-D convolution by several experiments.

## 1. はじめに

近年，自然言語処理における言語の獲得にはコーパスを使用した RNN による文章生成[Sutskever 2014, Auli 2013]，単語をベクトル化し類義語の探索を行う Word2vec[Mikolov 2013]，k次元の単語ベクトルを入力とした CNN による文章分類[Kim 2014]，文字レベルで特徴量を設定し 1-D Convolution による文章上の感情やカテゴリごとへのクラス分類[Zhang 2016]などが行われている．さらに，人工言語を取り扱った研究では，RNN を用いて質問文から論理演算子を導く処理[Neelakantan 2016]，2 つの 9 ケタの数字を足し合わせる処理[Zaremba 2015]，Log-bilinear Tree-Traversal Models を使用した C#言語の自動生成[Maddison 2015]などがある．

本研究では 1-D Convolution による文字レベルでの人工言語を学習し，変数名や関数名の出現に関して，RNN による学習を行う．これにより，スペルミスなどによるコンパイラエラーを引き起こすプログラムに対し，学習したモデルに通すことでエラーを取り除くことができる．今回は，1-D Convolution より符号化した 1 文字に対する学習を行う．そして，学習したモデルに対して符号化した文字を入力としたときの認識精度を確認する．

## 2. 人工言語の獲得

本システムでは，本学のプログラミング授業の際に学生が作成した C 言語プログラムに対し 1-of-k 符号化を行なったものを学習データとする．さらに，文字と単語の分類問題を 1-D Convolution より解き，語順に対する回帰問題は Recurrent Neural Network(RNN)を用いて学習する．

### 2.1 人工言語の学習

本研究では，ubuntu16.04 LST 上で深層学習フレームワーク chainer を使用する．学習するデータは本学のプログラミングの授業で学生が作成した C 言語プログラムを用いる．

#### (1) 学習データの符号化

学習データは 1-of-k 符号化したものを使用する．対象となる文字は大小のアルファベット(52 次元)，数字(10 次元)，プログラムで使用される記号(33 次元)，タブと改行を含めた 97 の要素をもつ行列となる．また，空白(スペース)または符号化していない文字はすべて無視されるもの(零行列)として表現する．表 1 に 1-of-k 符号化の対象となる文字を示す．

表 1 符号化の対象となる文字

| | |
|---|---|
| 文字 | abcdefghijklmnopqrstuvwxyz<br>ABCDEFGHIJKLMNOPQRSTUVWXYZ |
| 数字 | 0123456789 |
| 記号 | -,;.!?:'"/\|_@#$%^&*`~+-=<>()[]{} + “\t”, “\n” |

#### (2) 1-D Convolution による文字の学習

符号化した学習データを入力とし，1-D Convolution による畳み込みを行う．学習モデルは，畳み込み層と max pooling 層の組み合わせを 3 層，畳み込み層のみが 3 層，そして全結合層を含む 9 層で構成する．図 1 に符号化した文字，図 2 に本システムで使用する学習モデルを示す．

input feature
"a" = [1. 0. 0. 0. ... 0. 0. 0.]

図1 文字の 1-of-k 符号化

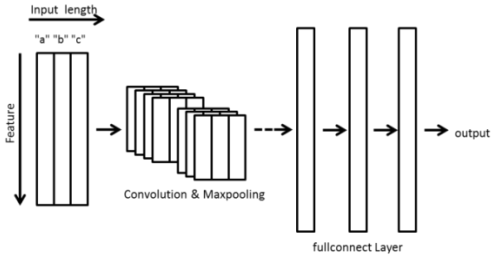


図 2 入力文字列データ畳み込みモデル

文字と単語の 1-D Convolution 関数は以下の式(1)で求められる．

$$h(y) = \sum_{x=1}^{k} f(x) \cdot g(y \cdot d - x + c) \qquad (1)$$

畳み込みの式は，カーネル関数$f(x)$，入力$g(x)$，ストライド$d$，オフセット定数$c$から成る．ここで，$c = k - d + 1$である．

また，1-D Convolution 関数より畳み込まれた文字の特徴量を max pooling 関数(式(2))の入力値$g(x)$とし，計算を行う．

$$h(y) = max \sum_{x=1}^{k} g(y \cdot d - x + c) \qquad (2)$$

連絡先：河原崎徳之，神奈川工科大学 ロボットメカトロニクス学科，神奈川県厚木市下荻野 1030，kawara@rm.kanagawa-it.ac.jp

式(2)の変数は式(1)と同様である.

また, 畳み込まれたデータは 3 層の全結合層を通し, log Softmax 関数による文字, 単語の分類を行う.

### (3) RNN による文字列の学習

RNN では, 1-D Convolution より得られた特徴量より単語内の文字列の順序, または文中の単語の順序についての回帰問題を解く. 図 3 に 1-D Convolution より得られた特徴量を入力とした RNN のモデルを示す.

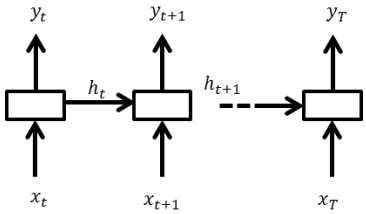


図 3 RNN モデル

## 2.2 生成したプログラムのコンパイル

生成したモデルに対して, 本校のプログラミングの授業で学生が作成した C 言語プログラムの中でコンパイルが通らないものをテストデータとする. このテストデータを図 3 のモデルに通し, エラーとなる文字列が修正され, コンパイラを通すことができるかどうかを確認する.

## 3. 文字の認識実験

本システムで符号化の対象とした 97 文字に対して, 1-D Convolution を用いて認識実験を行う. 使用した訓練データとして, 対象の 97 文字分のデータを 1-of-k 符号化したものを 60 個用意し, epoch 数を 100 とした. これらよりランダムに選択したデータ 128 個をミニバッチとして学習した. 検証データは訓練データで使用したものと同じものを使用する. 使用した最適化手法は SGD で学習率 5e-3, Momentum を 0.9, 重み減衰 1e-5, 文字の分類には log Softmax 関数, 損失関数には cross entropy を使用した. そして, すべての全結合層で Dropout を行う確率が 0.0, 0.1, 0.5 のとき, 1 文字の特徴量に対する学習損失と認識精度に変化があるかを確認する.

### (1) 符号化した文字の学習による損失

訓練データを用いた学習モデルの学習の損失を図 4 に示す. 図 4 のグラフは,横軸が訓練回数, 縦軸が学習で損失する値を示す. このグラフより, 1-D Convolution による畳み込みを行った特徴量に対し全結合層での Dropout を行う確率が大きいほど学習の損失が大きくなることがわかる.


図 4 Dropout の確率による訓練データの学習損失

### (2) 符号化した文字の認識精度

検証データによる学習モデルによる文字の認識精度を図 5 に示す. 図 5 のグラフは横軸を検証回数, 縦軸が文字の認識精度を示す. このグラフより(1)と同様, Dropout を行う確率が低いほど 1 文字に対する認識精度が高くなることがわかる. 特に, Dropout が 0.0 のとき認識精度は 100%近くであった.

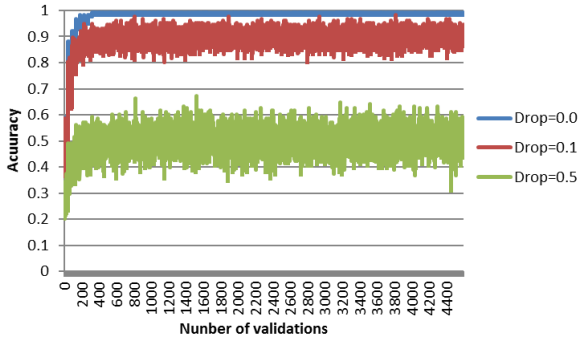


図 5 Dropout の確率による検証データの認識精度

## 4. おわりに

本研究では, 1-D Convolution を用いて 1-of-k 符号化した文字の認識実験を行った. 実験結果より, 全結合層で Dropout を行う確率を適用しない場合, 1-D Convolution により畳み込まれた文字の特徴量に対して認識する精度が最も高くなることが示された. 今後は, さらに単語を 1 文字レベルで符号化, または符号化した文字列を 1 つの行列とした特徴量より RNN を通して学習を行い, 人工言語内で用いられる変数や関数名よりコンパイラエラーを引き起こすスペルミスに対する修正を行えるモデルを生成したい.

## 参考文献


[Sutskever 2014] Ilya Sutskever, Oriol Vinyals, Quoc V. Le: Sequence to Sequence Learning with Neural Networks, NIPS 2014, 2014.

[Auli 2013] Michael Auli, Michael Galley, Chris Quirk, Geoffrey Zweig: Language and Translation Modeling with Recurrent Neural Networks, NIPS 2013, 2013.

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

[Zhang 2015] Xiang Zhang, Junbo Zhao, Yann LeCun: Character-level Convolution Networks for Text Classification, NIPS 2015, 2015.

[Kim 2014] Yoon Kim: Convolution Neural Networks for Sentence Classification, EMLNP 2014, 2014.

[Neelakantan 2016] Arvind Neelakantan, Quoc V. Le, Ilya Sutskever: NEURAL PROGRAMMER: INDUCTING LATENT PROGRAMS WITH GRADIENT DESCENT, ICLR 2016, 2016.

[Zaremba 2015] Wojciech Zaremba, Ilya Sutskever: LEARNING TO EXECUTE, ICLR2015, 2015.

[Maddison 2015s] Chris J. Maddison, Daniel Tarlow: Structured Generative Models of Natural Source Code, ICLR2015, 2015.