C-022

ハードウェア化に適した近似関数の導入による RNN 回路のリソース削減と低消費電力化

A Low Power RNN Hardware Architecture Based on Approximate Functions to Reduce Hardware Resources

村田大智†, **望月香那**†, **黒田幸作**†, **廣瀬哲也**†, **黒木修隆**†, **沼 昌宏**†
Daichi Murata†, Kana Mochizuki†, Kousaku Kuroda†,
Tetsuya Hirose†, Nobutaka Kuroki†, and Masahiro Numa†

1．まえがき

近年，画像認識や自然言語処理等を高精度に実現するための技術として，ニューラルネットワークに注目が集まっている [1] 。しかし，ニューラルネットワークは，膨大な処理時間が必要となるため，一般の CPU（Central Processing Unit）上でのソフトウェアによる処理は困難である。そこで，アクセラレータを用いて処理を高速化する必要がある。このアクセラレータとして，従来は GPGPU（General Purpose Graphic Processing Unit）が利用されることが一般的であった。しかし，GPGPU は汎用プロセッサであるため，処理に無駄が多く，消費電力が大きい [1] 。このため，常時稼働が必須となる大規模サーバ向けアクセラレータとして，GPGPU は不向きである。そこで，GPGPU の約 10 分の 1 の消費電力で動作可能な FPGA（Field-Programmable Gate Array）上にニューラルネットワークの処理を専用に行う回路を実装し，低消費電力化を実現する研究が盛んに行われている [1], [2]。

本稿では，ニューラルネットワークの 1 種である RNN（Recurrent Neural Network）[3] を,専用回路により実現するアーキテクチャを提案する。具体的には，専用回路による実現が困難なシグモイド関数や双曲線正接関数を 1 次関数で近似することで，RNN 回路の実現に必要となるリソースを削減すると同時に，低消費電力化を図る。

2．関連研究

2.1 RNN

RNN は，時系列情報の学習に有効なニューラルネットワークであり，自然言語処理に応用される [3], [4] 。図 1 に RNN の処理フローを示す。ここで，x_t は時刻 t における入力データを示す。RNN は，時系列情報を記憶するための LSTM（Long Short-Term Memory）レイヤ，各 LSTM レイヤからの出力をまとめるための mean pooling，および識別器，学習器から構成されている。LSTM レイヤでは，全ての時系列情報の中で，重要度の高い情報のみを選別・記憶する。これにより，過去の全ての情報を記憶する場合と比較して，RNN 内に蓄積される情報量を削減し，情報量の過多による情報の消失や爆発を抑制することができる。その結果，比較的長い時系列情報も学習することが可能となる。

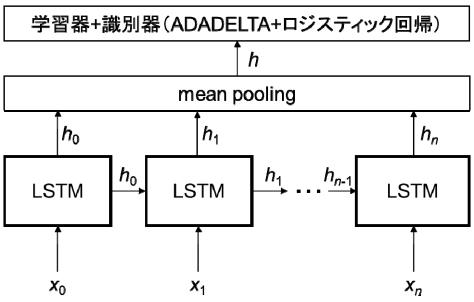


図 1 RNN の処理

2.2 LSTM レイヤ

図 2 に LSTM レイヤの処理フローを示す [3], [4] 。LSTM レイヤは，全ての時系列情報の中で重要度の高い情報のみを選別するための input gate，forget gate，output gate および情報の記憶を担当する memory cell で構成される。以下，具体的な LSTM レイヤの処理について述べる。ただし，x_t は時刻 t における入力データを，h_{t-1} は時刻 t–1 における LSTM レイヤの出力を，W, U は重み行列，b はバイアスベクトルを示す。

まず，memory cell によって記憶する情報の候補を，input modulation gate により

$$\tilde{C}_t = \tanh(W_c x_t + U_c h_{t-1} + b_c) \quad (1)$$

と算出する。また，input gate ならびに forget gate の出力を

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (2)$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (3)$$

により求める。ここで，時刻 t における memory cell の値 C_t は，

$$C_t = i_t \circ \tilde{C}_t + f_t \circ \tilde{C}_{t-1} \quad (4)$$

と表現できる。次に，output gate の出力を

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (5)$$

により求める。以上より，LSTM レイヤの出力 h_t は，

$$h_t = o_t \circ \tanh(C_t) \quad (6)$$

で表される。

2.3 問題点

前節で述べた LSTM レイヤは，双曲線正接関数 tanh や，シグモイド関数 σ 等，回路大規模化の要因となる演算が多用されている。このため，従来の LSTM アルゴリズムをもとに回路を設計すると，回路規模および消費電力の増大が問題となる。

3．提案手法

3.1 提案する LSTM アルゴリズム

本稿では，双曲線関数およびシグモイド関数を 1 次関数で近似することで，小規模回路で低消費電力な LSTM レイヤ回路を実現する手法を提案する。

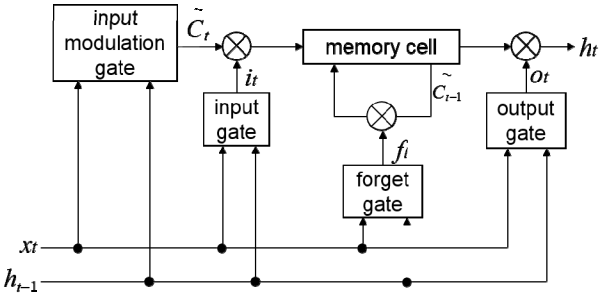


図 2 LSTM レイヤの処理

† 神戸大学, Kobe University

表 1　LSTM レイヤ回路の FPGA へのマッピング結果

リソース		搭載数	従来		提案		削減率 [%]	
			利用数	利用率 [%]	利用数	利用率 [%]		
演算器	LUT	3.04×10^5	2.37×10^4	7.80	2.68×10^3	0.88	88.7	88.6
	DSP	2.80×10^3	1.48×10^2	5.29	3.20×10	1.14	78.4	
メモリ	LUT	1.30×10^5	4.48×10^2	0.34	0.10×10	0.00	99.8	79.1
	Register	6.07×10^5	1.13×10^4	2.19	2.45×10^3	0.40	78.3	
	BRAM	1.03×10^3	0.25×10	0.24	0.25×10	0.24	0.0	

具体的には，シグモイド関数 y_σ を

$$y_\sigma = \min(1, \max(0, 0.125x + 0.5)) \quad (7)$$

で近似する。加えて，双曲線正接関数 $y_{\tanh}$ を

$$y_{\tanh} = 2\min(1, \max(0, 0.125x + 0.5)) - 1 \quad (8)$$

により近似する。

3.2　提案近似関数のハードウェア構成

図 3 に，提案する近似関数のハードウェア構成を示す。従来，シグモイド関数や双曲線関数を実現する際には，指数関数や除算等，回路大規模化の要因となる関数を複数組み合わせて構成する必要があった。一方，提案する近似関数は，回路での実現が容易なシフト演算や加減算，セレクタのみで実現できるため，回路規模と消費電力を大きく削減可能である。

4．シミュレーション評価

4.1　アルゴリズムの精度評価

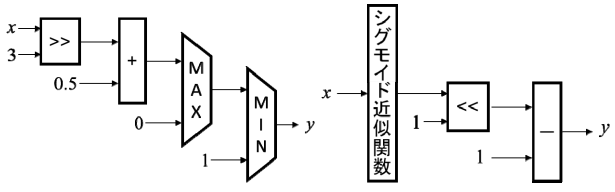
提案した近似関数を用いて RNN 全体を構成し，動作レベル・シミュレーションにより，RNN 精度を評価した。評価環境は Python 2.7.6，CUDA7.5，cuDNN v3，Theano 0.7，nVIDIA GeForce GTX TITAN X である。また，データセットには Large Movie Dataset [5] を用いた。

図 4 にシミュレーション結果を示す。近似関数を用いる提案手法において，RNN の精度低下は 0.012 ポイントに抑制され，精度低下は無視できる範囲に留まることを確認した。

4.2　FPGA へのマッピングによる評価

近似関数を用いる提案アーキテクチャと，近似関数を用いない従来アーキテクチャのそれぞれに関して，LSTM レイヤ 1 層分の回路を設計し，配置配線後の回路規模および消費電力に関して評価を行った結果を表 1 に示す。論理合成には Xilinx 社の Vivado 2015.4 を用いて，同社の VC707 評価ボードに搭載する FPGA（Virtex7 : XC7VX485T-2FFG1767C）にマッピングを行った。

まず回路実装に必要なリソース数に関して，近似関数を利用することで，演算器を 88.6%，メモリを 79.1% 削減することができた。これは，LSTM レイヤ回路から指数関数回路や除算器を省くことができたことに起因すると考えられる。次に消費電力に関して，提案アーキテクチャは従来アーキテクチャより 81.8% 低い消費電力で動作可能であることを確認した。これは，回路規模削減に付随する効果であると考えられる。



(a) シグモイド近似関数　　(b) 双曲線正接近似関数

図 3　提案する近似関数のハードウェア構成

5．まとめ

本稿では，FPGA を用いた低電力 RNN アクセラレータの構築に向けて，RNN の中核をなす LSTM レイヤに関して回路設計を行った。LSTM レイヤ内のシグモイド関数や双曲線関数を回路化が容易な近似関数に置き換えることで，LSTM レイヤを小規模かつ低消費電力な回路で実現する手法を提案した。

提案した近似関数を用いて LSTM レイヤ回路を設計・評価した結果，演算器を 88.6%，メモリを 79.1%，消費電力を 81.8% 削減する効果を得た。また，近似関数を用いることによる RNN の精度低下は，無視できる範囲に留まることを確認した。

今後の課題として，RNN 全体のハードウェア・アーキテクチャ設計と実装評価が挙げられる。

参考文献

[1] 中原啓貴，笹尾勤，“Nested RNS を用いた深層畳み込みニューラルネットワークに関して”，電子情報通信学会，vol. 115, no. 109, pp. 91-96, 2015 年 6 月.

[2] K. Ovtcharov, et.al, “Accelerating deep convolutional neural networks using specialized hardware,” Microsoft Research, 2015.

[3] F. Gers, “Long short-term memory in recurrent neural networks,” Diss. Universität Hannover, 2001.

[4] J. Donahue, et.al, “Long-term Recurrent Convolutional Networks for Visual Recognition and Description,” IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[5] Large Movie Review Dataset, http://ai.stanford.edu/~amaas/data/sentiment/ .

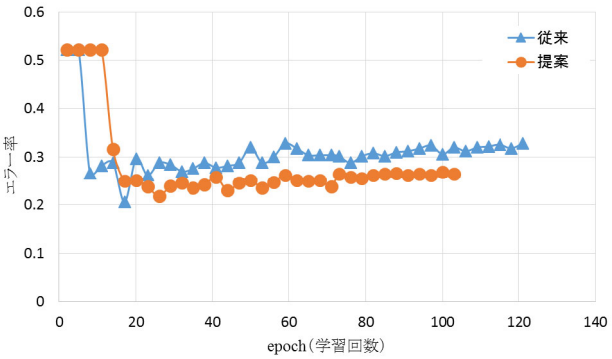


図 4　RNN 精度に関する比較評価結果