Long short-term memory for radio frequency spectral prediction and its real-time FPGA implementation

Siddhartha[1], Yee Hui Lee[1], Duncan J.M. Moss[1], Julian Faraone[1], Perry Blackmore[2], Daniel Salmond[2], David Boland[1], and Philip H.W. Leong[1]

[1] School of Electrical and Information Engineering
[2] Defence Science and Technology Group

Australian Government
Department of Defence
Science and Technology

THE UNIVERSITY OF SYDNEY

Overview

- Introduction
- Long short-term memory (LSTM)
- Implementation
- Results
- Conclusion

Motivation and Aims

› Motivation

- Highly dynamic and complex environments pose a challenge for current tactical radios
- LSTMs have been extremely successful at difficult tasks such as speech recognition and machine translation
- LSTM suitability for real-time radio applications not well studied
- Can we effectively use ML in the next generation of tactical radios?

› Aims

- Apply LSTMs to spectral prediction in radio frequency signals
- Determine utility over conventional time-series prediction schemes
- Understand accuracy/area trade-offs for fixed-point FPGA implementations
- Understand latency achievable with FPGA implementations

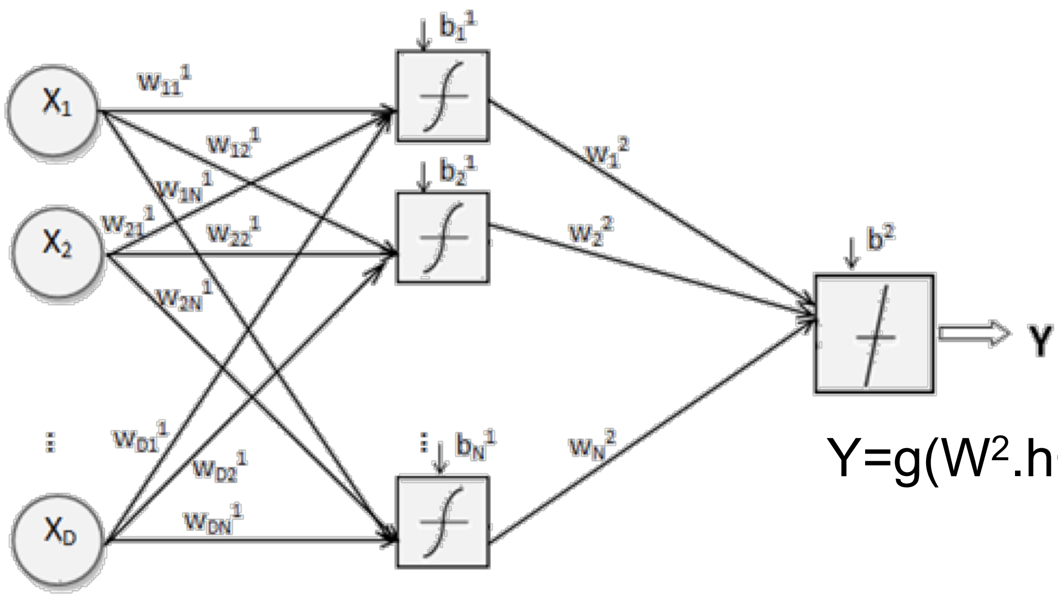
FPGAs

- Field programmable gate arrays (FPGAs) are COTS, user-customisable integrated circuits
- Unique benefits over uP/DSP/GPU/ASIC due to
  - **E**xploration – easily try different ideas to find best solution
  - **P**arallelism – so we can arrive at an answer faster
  - **I**ntegration – so interfaces are not a bottleneck
  - **C**ustomisation – problem-specific designs to improve efficiency

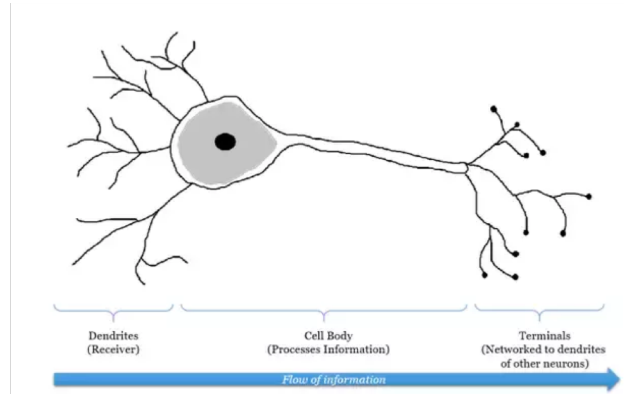
Overview

- Introduction
- Long short-term memory (LSTM)
- Implementation
- Results
- Conclusion

Feedforward Neural Network



(Matheus, 2016)

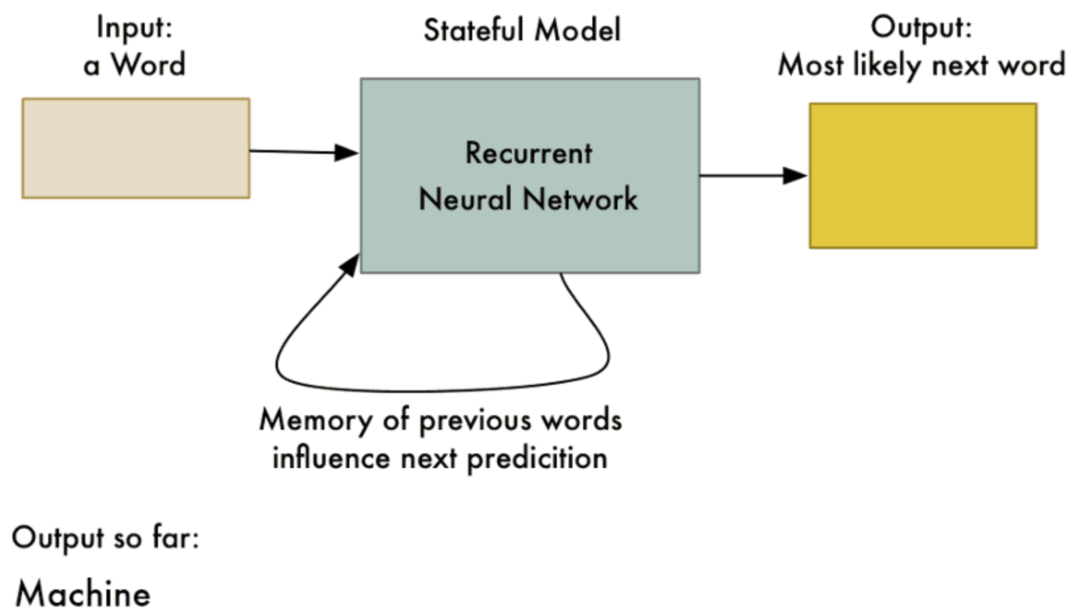


$$Y=g(W^2.h+b^2)$$

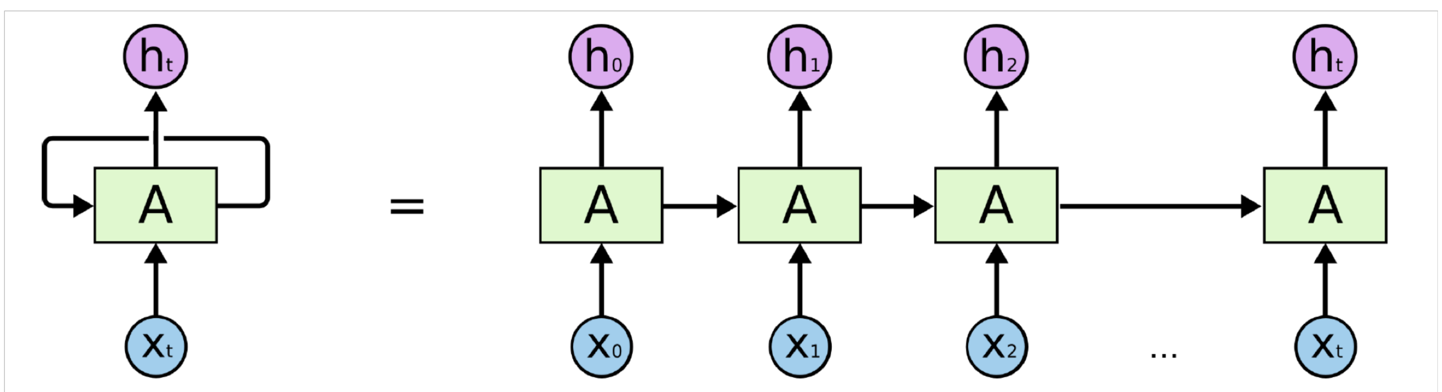
$$h=f(W^1.X+b^1)$$

No state → Can't deal with sequential data

Recurrent Neural Networks



Can't deal with large gaps

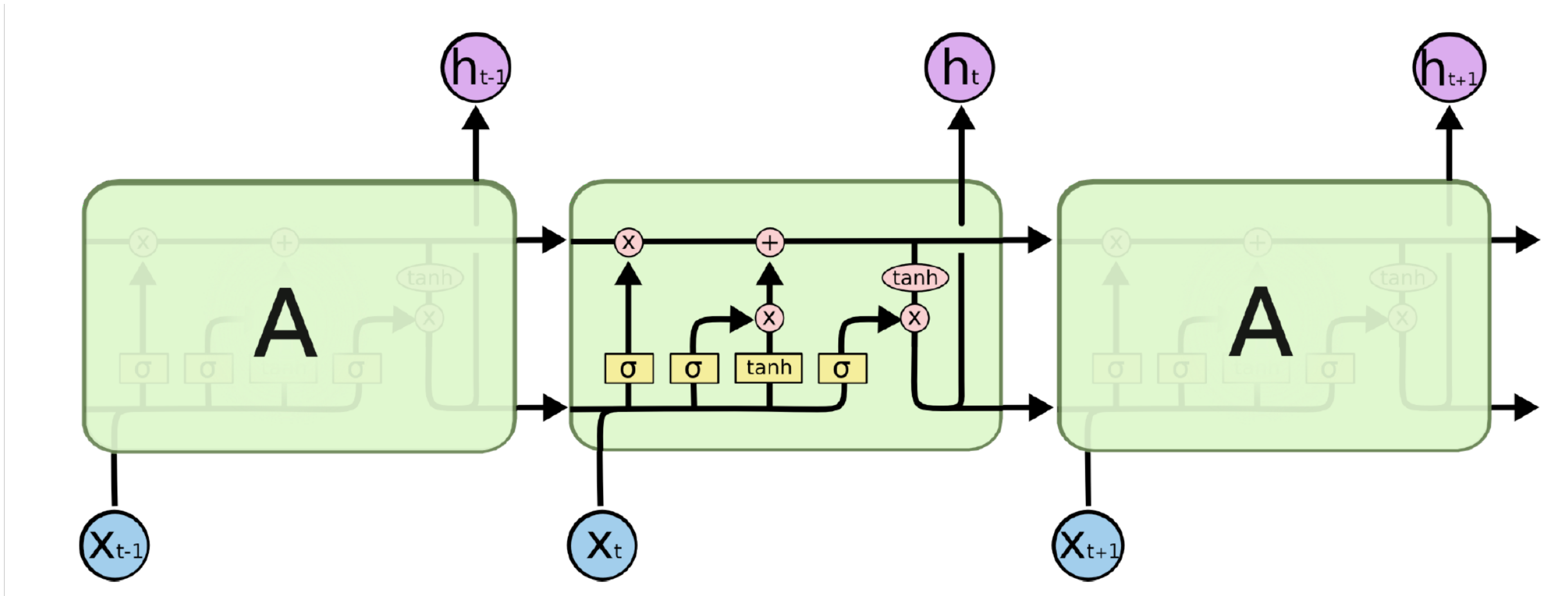


An unrolled recurrent neural network.

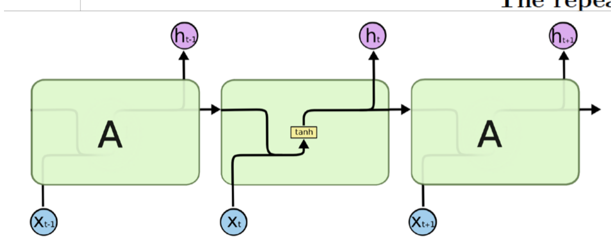
Source: colah's blog (with permission)

Long short-term Memory

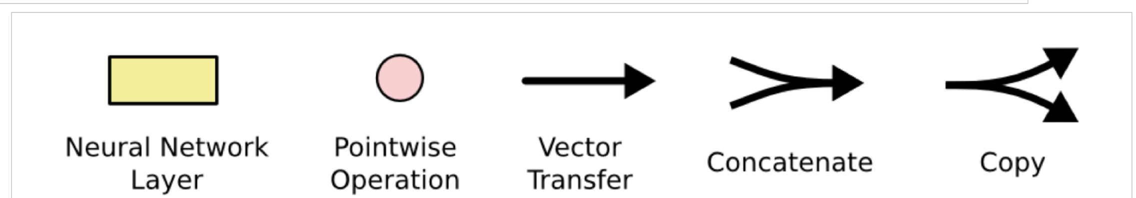
Long Short-Term Memory is a type of **gated** Recurrent Neural Network (RNN)
Proposed by Hocreiter and Schmidhuber in 1997



The repeating module in an LSTM contains four interacting layers.



The repeating module in a standard RNN contains a single layer.



Source: colah's blog (with permission)

LSTM Mathematical Description

- LSTM

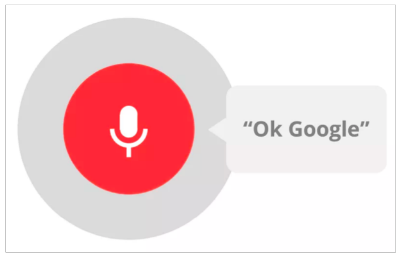
$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \text{sigm} \\ \text{sigm} \\ \text{sigm} \\ \tanh \end{pmatrix} T^{l}_{(n_{l-1}+n_l),(4n_l)} \begin{pmatrix} h_t^{l-1} \\ h_{t-1}^{l} \end{pmatrix}$$

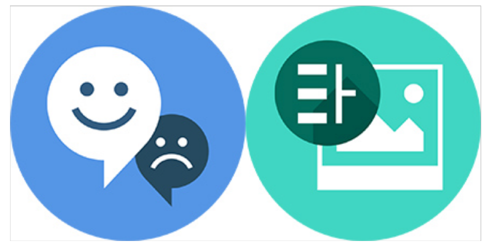
$$\begin{aligned} c_t^l &= f \odot c_{t-1}^l + i \odot g \\ h_l^t &= o \odot \tanh(c_t^l) \end{aligned}$$

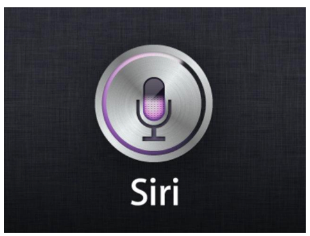
- Followed by a single linear fully connected layer

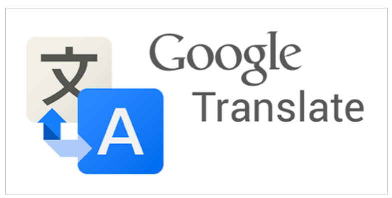
$$f_t = T^{L+1}_{n_L,n_L} h_t^L$$

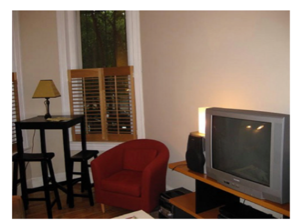
Applications





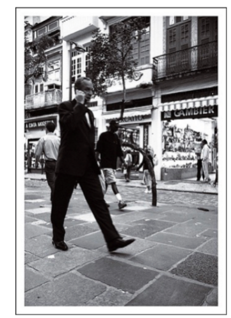






↑ a living room with a couch and a television

↑ a man riding a bike on a beach

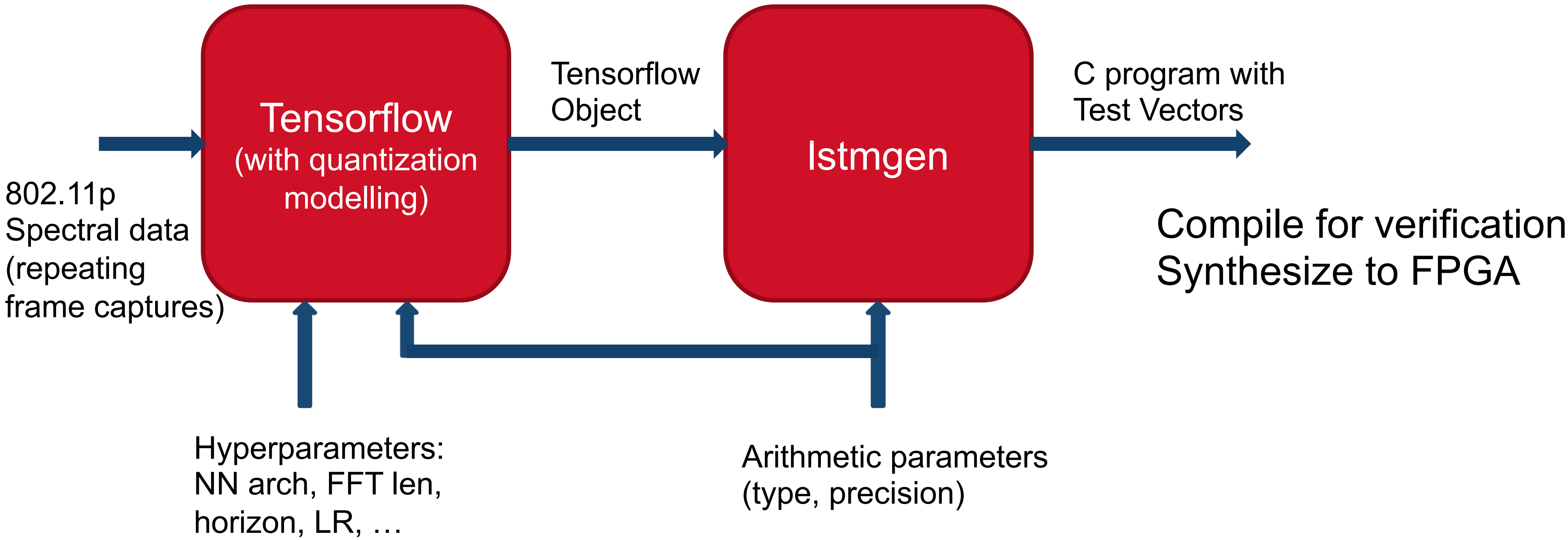


a man is walking down the street with a suitcase ↗

Overview

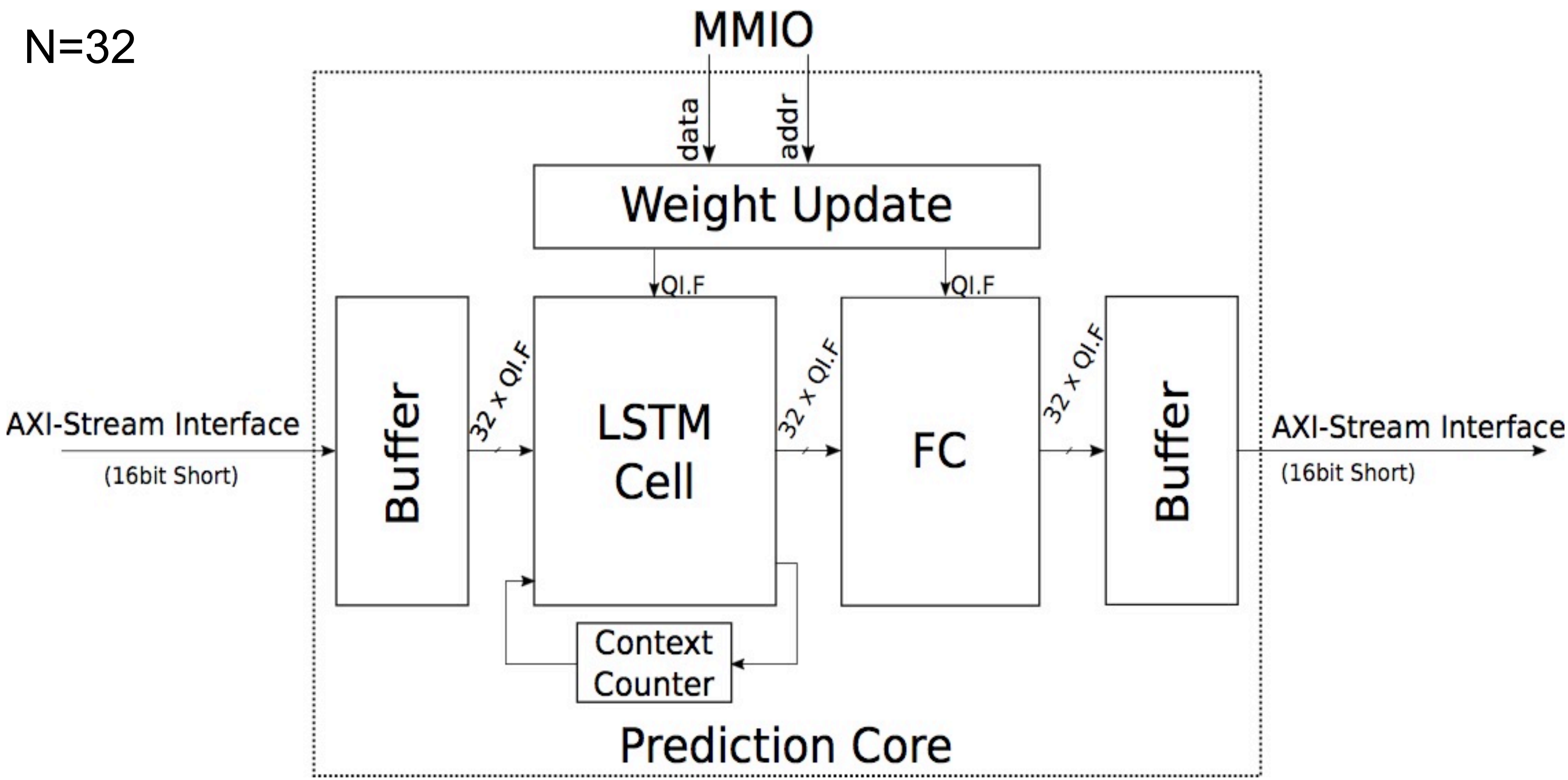
- Introduction
- Long short-term memory (LSTM)
- Implementation
- Results
- Conclusion

Design Flow


802.11p
Spectral data
(repeating
frame captures)
Tensorflow
(with quantization
modelling)
Tensorflow
Object
lstmgen
C program with
Test Vectors
Compile for verification
Synthesize to FPGA
Hyperparameters:
NN arch, FFT len,
horizon, LR, …
Arithmetic parameters
(type, precision)

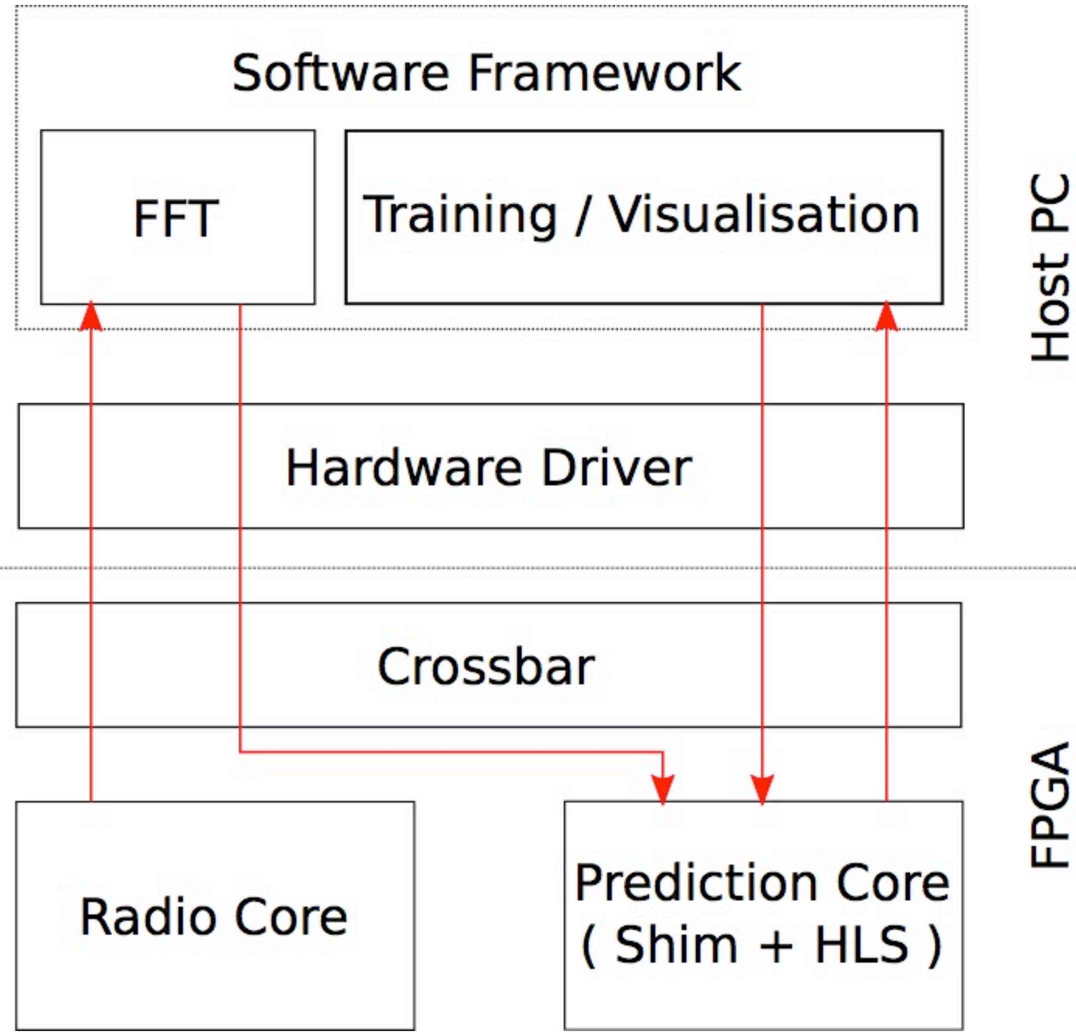
LSTM Core Architecture

N=32



System Architecture

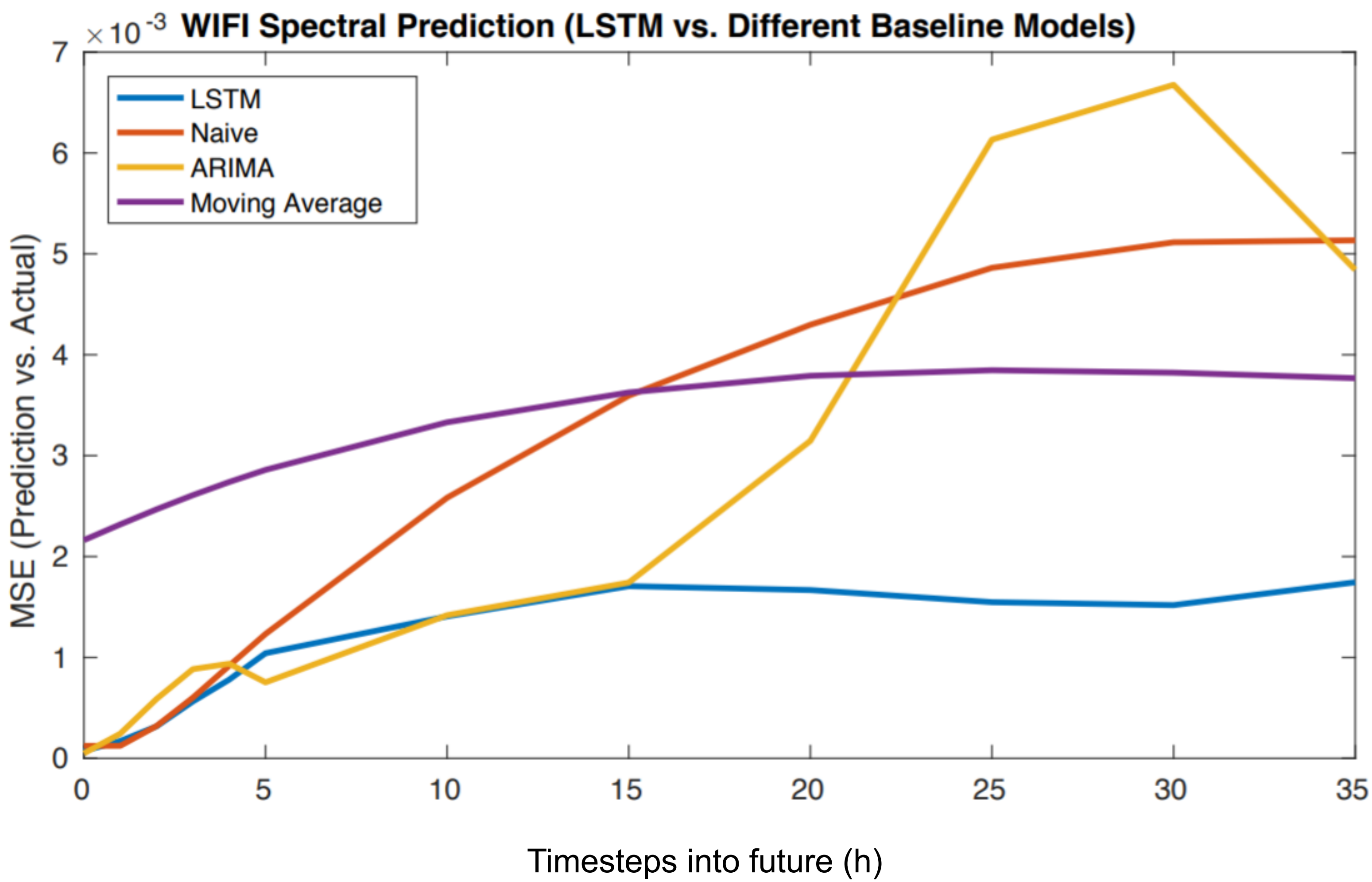
- Implemented on Ettus X310
- Software
  - GNU Radio integration to manage data movement
  - Offline LSTM training
- Hardware Acceleration
  - RFNoC framework
  - Prediction Core on FPGA



Overview

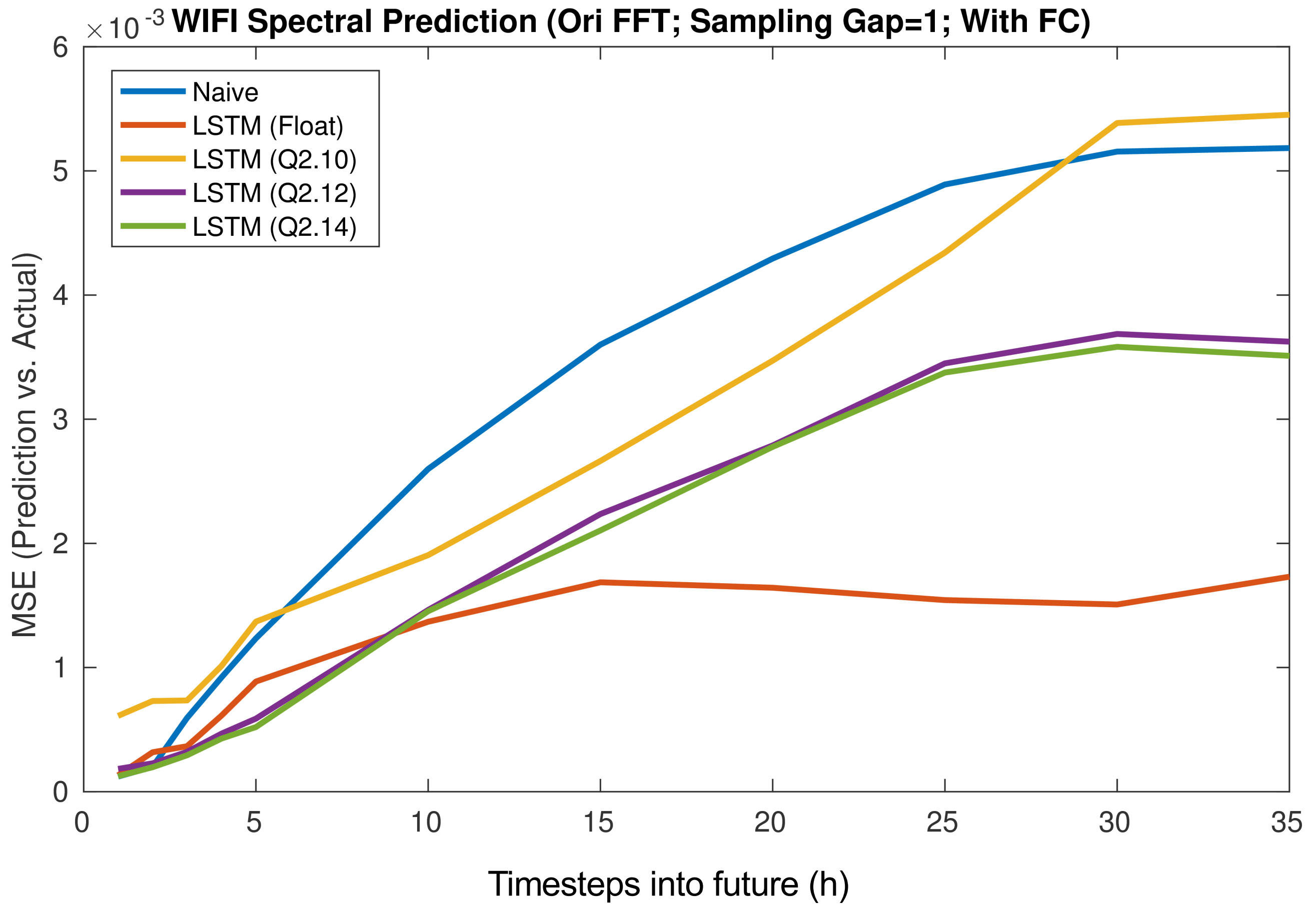
- Introduction
- Long short-term memory (LSTM)
- Implementation
- Results
- Conclusion

Floating-Point Accuracy



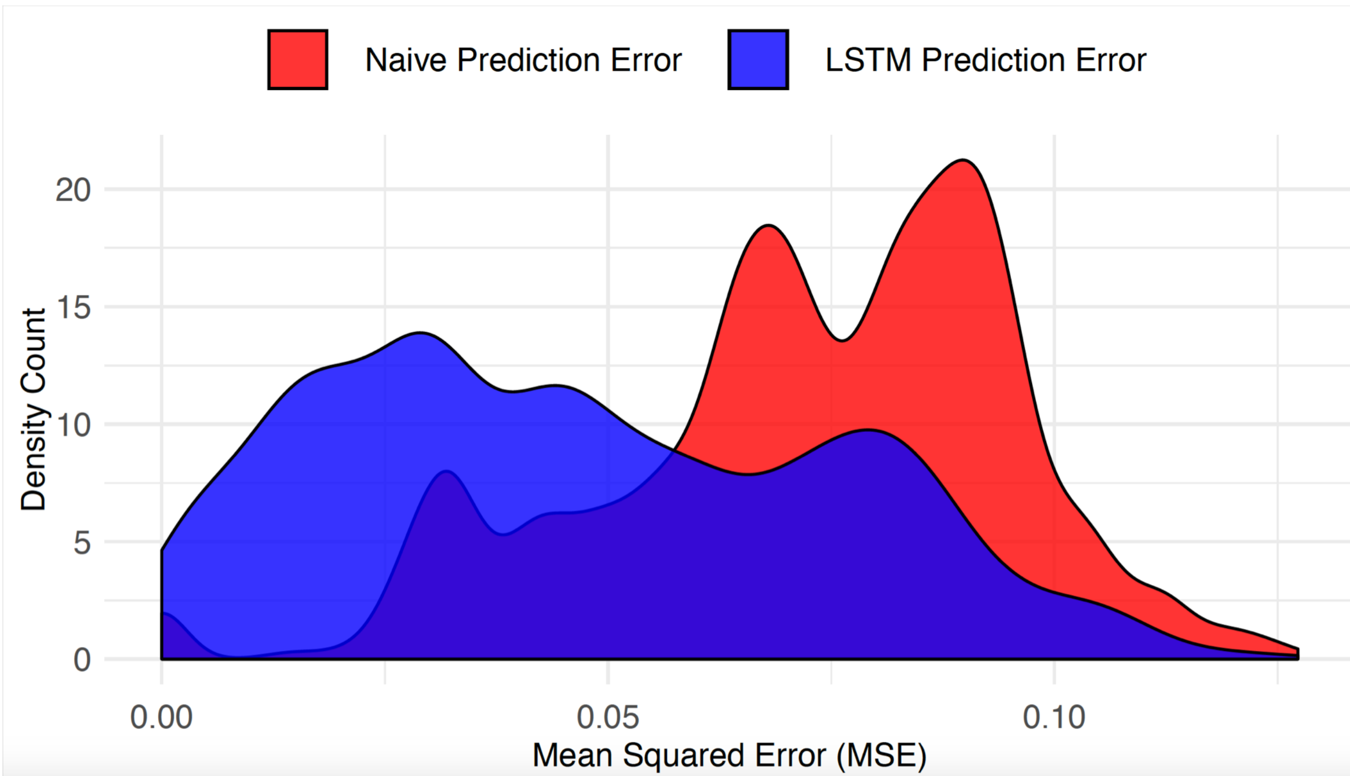
Fixed-point Implementation

- Fixed-point implementations have lower latency
- Q2.12 needed to preserve numerical accuracy



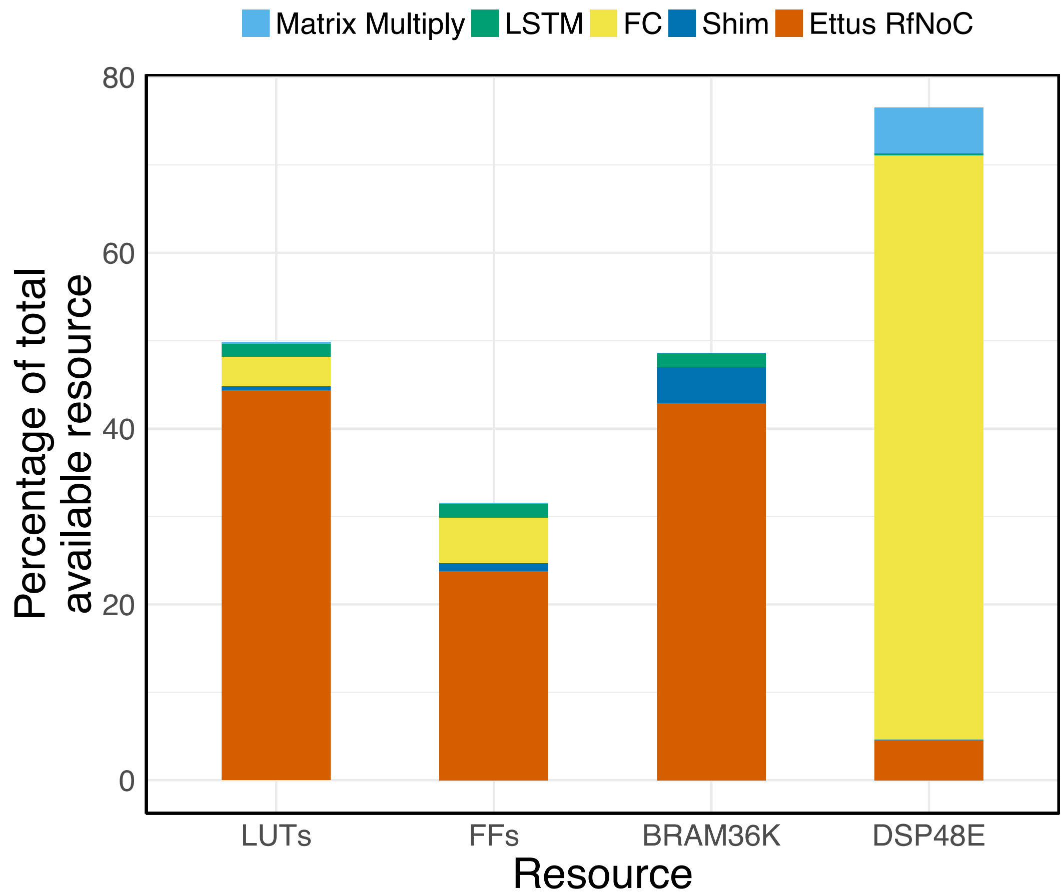
Prediction Accuracy

- N=32 history, h=4 prediction horizon
- Accuracy measured as the mean-squared error loss from true value
- LSTM gives better predictions than conventional approaches



Resource utilization

- C-code synthesised to Kintex-7 XC7K410T FPGA for Ettus X310
  - Achieves 4.3 μs latency (32 inputs and outputs)
- Limited by DSPs (~80% of 1540 available)
  - FC layer is fully unrolled to reduce prediction latency
- Most logic resources and on-chip memory used by RFNoC framework
  - Could customize design to reduce footprint and allow larger/deeper networks
  - Kintex Ultrascale with 2x more DSPs are already available



Overview

- Introduction
- Long short-term memory (LSTM)
- Implementation
- Results
- Conclusion

Conclusion

- Described an LSTM module generator
  - Compatible with Tensorflow
  - Generates C programs of arbitrary size, topology and precision
  - Testable and synthesisable to efficient FPGA implementation
- Low-precision fixed point LSTM can achieve better spectral prediction accuracy than conventional approaches such as Naïve or ARIMA
- Real-time LSTM-based spectral prediction feasible
  - Input/output lengths of 32; Q2.12 implementation fits easily on Ettus X310 and achieves latency of 4.3 us
- Our future research will explore how such predictions can be used to improve tactical radios