CSE847 Project Proposal

An analysis and verification of the Fast Associative Memory method for RNNs

Eric Wayman
Department of Computer Science and Engineering
Michigan State University
waymane1@msu.edu

Adi Mathew
Department of Computer Science and Engineering
Michigan State University
mathewad1@msu.edu

ABSTRACT

This paper provides a sample of a LaTeX document which conforms, somewhat loosely, to the formatting guidelines for ACM SIG Proceedings.

KEYWORDS

ACM proceedings, LaTeX, text tagging

1 PROBLEM DESCRIPTION

Until recently, recurrent neural networks (RNNs), used for sequential processing, were limited by the fact that within-sequence memory was limited to short-term only (long-term memory was limited to between-sequence memory). The "fast weights" method introduced in the paper to be analyzed in this project (Ba et al. 2016a) addresses this limitation by providing the network with the capacity to store information about a given sequence during its duration (to be used during each step in the hidden layers). We will provide a full analysis and explanation of the methology, and replicate one of the empirical tests of the method, which compares its performance on an associative retrieval task to that of an iRNN and an LSTM.

2 SURVEY OF PRIOR WORK

Recurrent neural networks (RNNs) are well-suited for learning from sequential data since weights are shared among different stages of the sequence (Goodfellow et al. 2016, p. 373). One major issue in training RNNs with many layers was that the gradients (of the error with respect to a particular weight) end up becoming very large or small (Schmidhuber 2015, p. 16). This was overcome by the introduction of the long short-term memory network (LSTM RNN), whose activation function has a constant derivative and thus does not explode or vanish (Schmidhuber 2015, p. 19).

{Next: explain weakness of LSTMs and how introduction of fast associative memory addresses that weakness.}

Hopfield nets, associative memory: (Mackay 2003)

Layer normalization: (Ba et al. 2016b)

Grid search: (Goodfellow et al. 2016)

Adam optimizer: (Kingma and Ba 2014)

IRNN definition: (Talathi and Vartak 2015)

3 PRELIMINARY PLAN

Our term paper will first present the fast associative memory methodology and place it in the context of methods that led to its development. We will provide an extended description and derivation of the methodology for the purpose of verifying its properties. We will also replicate section 4.1 of the paper, which compares the fast associative memory method's performance on an associative retrieval task with that of an Identity-RNN (iRNN) and LSTM.

Table 1: *Project timeline*

Week	Task	Deliverable
Week 1 ()		
Week 2 ()		
Week 3 ()		
Week 4 ()		
Week 5 ()		
Week 6 ()		
Week 7 ()		
Week 8 ()		
Week 9 ()		
Week 10 ()		

REFERENCES

Jimmy Ba, Geoffrey E Hinton, Volodymyr Mnih, Joel Z Leibo, and Catalin Ionescu. 2016a. Using Fast Weights to Attend to the Recent Past. In *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (Eds.). Curran Associates, Inc., 4331–4339. http://papers.nips.cc/paper/6057-using-fast-weights-to-attend-to-the-recent-past.pdf

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016b. Layer Normalization. (2016). arXiv:1607.06450

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. (2014). arXiv:arXiv:1412.6980 (published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015)

David J. C. Mackay. 2003. *InformationTheory, Inference, and Learning Algorithms.*

Jürgen Schmidhuber. 2015. Deep learning in neural networks: An overview. *Neural Networks* 61 (2015), 85–117. DOI:http://dx.doi.org/10.1016/j.neunet.2014.09.003

Sachin S. Talathi and Aniket Vartak. 2015. Improving performance of recurrent neural network with relu nonlinearity. (2015). arXiv:arXiv:1511.03771