

# PDI – Relatório de Progresso

*José Pedro Castro Fonseca*

November 28, 2015

## 1 Análise Introdutória do Problema a Tratar

Esta Tese de Dissertação consistirá no projecto e implementação de um Acelerador de Hardware reconfigurável – baseado em FPGA<sup>1</sup> – para estruturas de Aprendizagem Computacional, nomeadamente **Redes Neurais Recursivas**.

### 1.1 – Contexto

As **Redes Neurais Artificiais** são um dos algoritmos de classificação mais populares do mundo da Inteligência Artificial. Tal como o nome sugere, são estruturas cuja operação é inspirada na forma como os blocos constituintes do nosso cérebro, os neurónios, operam. Apesar da sua elevada performance, um dos seus *shortcomings* é o facto de não conseguir reter informação de eventos anteriores na sua decisão actual, i.e. perante um vector de entrada  $x_i$ , a decisão sobre o vector de saída  $y_i$  não possui influência directa de entradas anteriores  $x_{i-1}, x_{i-2}, \dots$ , sendo assim desajustada para o processamento de séries temporais de dados, em que entradas actuais têm uma elevada dependência informacional e sintática de entradas anteriores, como a análise de texto, sinais determinísticos e a predição de novos valores dessas séries.

Uma das formas de melhorar o aspecto anterior, foi a introdução nos anos 80 de **elementos de memória**, sendo isto feito através do *feedback* das camadas exteriores de decisão para as camadas interiores, ou escondidas. A estrutura que utilizarei neste trabalho serão as *Long Short-Term Memory Networks* [1], propostas por Hochreiter et al. em 1997, e que desde então é uma das técnicas *state-of-the-art* na área da Aprendizagem Profunda (*Deep Learning*), tendo obtido a melhor classificação de sempre no reconhecimento de caligrafia [2] em 2009, reconhecimento de fala [3], entre outros.

### 1.2 – Motivação

Todas estas técnicas são geralmente implementadas em Processadores Convencionais, fazendo uso de linguagens de alto ou baixo nível. Desta forma, todo o paralelismo real inerente às estruturas de cálculo fica limitado ao número de *cores* físicos presentes na máquina, que hoje em dia não excedem os 4 ou 8 nos dispositivos convencionais usados pela maioria das pessoas (Computadores Pessoais ou dispositivos móveis). Assim, de forma a paralelizar completamente a estrutura e, por conseguinte, aumentar consideravelmente a performance, uma solução é a utilização de uma GPU's ou FPGA, permitindo a realização de aprendizagens cada vez mais complexas, e cada vez mais depressa. Além disso, quando a taxa de amostragem de dados de entrada é muito elevada, a solução convencional de CPU não é escalável e apresenta sérias limitações.

Apesar de já haver trabalho de adaptação de outros algoritmos de aprendizagem para Hardware, a área do *Deep Learning* ainda é um campo pouco explorado no Hardware, sendo que no caso específico de LSTM, ainda existe apenas

---

<sup>1</sup>Field-Programmable Gate Array, um tipo de Circuitos Integrados reconfiguráveis em Software, cuja estrutura é **descrita** usando linguagens como Verilog ou VHDL

---

um artigo [4], submetido ainda este mês, que explora uma primeira implementação desta técnica em FPGA para a geração automática de texto.

Ainda que os CPU's convencionais sejam insubstituíveis para as funções normais a que geralmente estão afectos, é possível que possam haver co-processadores baseados nestas estruturas que acelerem certas tarefas de reconhecimento automático, sem comprometer a performance do sistema global. Além disso, foi provado que as Redes Neurais Recorrentes são *Turing Complete* [5][6], significando, por isso, que podem simular o funcionamento de qualquer programa que possa ser executado num CPU convencional.

### 1.3 - Objectivos

Assim, tendo como base o trabalho apresentado em [4], a ideia é construir um sistema que, além daquele, permita uma reconfiguração completa, em tempo real, da topologia da rede, e a substituição das células básicas por outras com variações na sua estrutura.

Será feito, em primeiro lugar, um modelo conceptual do sistema numa linguagem de alto-nível que permita validar o funcionamento dos IP Cores que forem posteriormente desenvolvidos em Verilog, e que depois serão sintetizados para a plataforma-alvo FPGA. Além disso, será utilizado um ou dois *Benchmarks* conhecidos para, por um lado, comparar o desempenho do sistema FPGA versus a implementação em CPU e, por outro, comparar o desempenho da minha implementação com outros desempenhos de referência. Seria igualmente interessante escolher uma aplicação-tipo deste tipo de Redes, e testá-la na minha implementação, para poder observar o seu funcionamento prático.

## 2 Trabalho Realizado

Nesta secção é feita uma pequena demonstração das interações com o Orientador, assim como os eventos anteriores à atribuição do tema de dissertação que motivaram a proposta do mesmo.

### 2.1 - Pessoas Envolvidas na Dissertação

Para além de mim, o Candidato ao grau de Mestre, incluem-se duas pessoas, nomeadamente

- **Orientador** – O [Professor João Canas Ferreira](#), professor Auxiliar do Departamento de Engenharia Electrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto, e meu antigo docente da UC de [Projecto de Circuitos VLSI](#).
- **Co-Orientador** – O [Ivo Timóteo](#), candidato a PhD em Ciência de Computadores, na área de Inteligência Artificial, na Universidade de Cambridge, Reino Unido.

### 2.2 - Instâncias de Interação com o Orientador

As principais instâncias de interação com o orientador resumem-se nos seguintes pontos.

- **Sondagem dos Temas Disponíveis (4 de Setembro de 2015)** – Após me ter inscrito à UC de [Aprendizagem Computacional](#), comecei a navegar um pouco pelo mundo das aplicações do Machine Learning e da forma como este pode ser acelerado com estruturas de Hardware dedicadas, ao invés de utilizar Processadores convencionais. Adicionalmente, colegas de anos anteriores sugeriram-me que o Prof. João Canas Ferreira poderia ter algum trabalho disponível nessa área. Assim, enviei um email ao Prof. João Canas Ferreira para sondar os temas que teria disponíveis nesta área, e obtive resposta positiva. Combinamos, então, uma reunião Skype em que discutiríamos com mais detalhe esta hipótese, no dia 15 de Setembro.
- **Reunião por Skype (15 de Setembro de 2015)** – Nesta reunião, o Prof. Canas deu-me a conhecer alguns projectos em concreto que tinha em mente, e a abordagem ao problema que deveria ser seguida. Acordamos, então, que ele iria submeter uma [Proposta de Tema de Dissertação](#), mas não seria imediatamente alocada para mim, dado que eu gostaria de dar uma vista de olhos pelas restantes Temas, e só depois tomar uma decisão final e porque a minha média me permitiria escolher o tema em concurso normal, sem o problema de eu ser afastado pelo processo de seriação.

- 
- **Envio da Proposta de Dissertação (17 de Setembro de 2015)** – Envio, por email, da proposta de dissertação redigida pelo Prof. Canas, para minha análise. Esta proposta é uma indicação geral do trabalho a ser desenvolvido, sendo que o algoritmo específico e o equipamento utilizado poderiam sofrer modificações com o desenrolar da pesquisa.
  - **Submissão da minha lista de preferências no Concurso Normal (27 de Setembro de 2015)** – Neste dia, submeto a minha lista de preferências de escolha de tema de dissertação, em que me decido definitivamente por este tema. Informei, por email, o Prof. Canas da mesma escolha. Sugiro a possibilidade de se incluir como co-orientador o Ivo Timóteo.
  - **Reunião Presencial (5 de Novembro de 2015)** – Discussão do plano de trabalhos, dos objectivos a serem atingidos e das metodologias a serem desenvolvidas. Nesta reunião, apresentei também alguns resultados preliminares da minha primeira pesquisa sobre a matéria, e o Orientador sugere, adicionalmente, a incorporação dos eventuais resultados da dissertação num projecto em desenvolvimento no INESC, como uma possibilidade.
  - **Reunião Presencial (26 de Novembro de 2015)** – Neste dia, apresento uma versão preliminar deste relatório ao Orientador para discussão, assim como uma primeira ideia de trabalho, explicada na Secção 1.

### 2.3 – Sequência Temporal de Trabalhos

Toda a frequência da UC de Aprendizagem de Computacional foi uma oportunidade de me inteirar das diferentes técnicas da área. Após este conhecimento básico, passei uma boa parte do mês de Outubro a fazer uma pesquisa sobre implementações destas técnicas em Hardware, e descobri que as técnicas básicas iniciais de SVM e Médias k-NN – presentes na proposta inicial – são já bastante trabalhadas, sendo pertinente escolher uma nova área de foco. Assim, depois da primeira reunião presencial com o Orientador, foquei-me durante o mês de Novembro em encontrar técnicas que, apesar de serem estado-da-arte, não tivessem sido exploradas na vertente de implementação em hardware, ou em caso positivo, isso estivesse ainda no começo. Assim, depois de uma pesquisa bibliográfica nos sites [Engineering Village](#), fui conduzido a estas técnicas de Redes Neurais Recorrentes. Procurei, também, informação de alto nível no site do Prof. Jürgen Schmidhuber (*Università della Svizzera italiana*, Suíça), um dos autores deste algoritmo. Após esta fase preliminar de recolha de informação, e de ter trocado impressões com o Orientador e o Co-Orientador, sintetizei toda essa informação neste Relatório de Progresso.

## 3 Pontos a Concretizar no Relatório Seguinte

Para o Relatório Final de PDI, é necessário fazer um levantamento exaustivo de todo o estado da arte neste tipo de redes, bem como uma pesquisa bibliográfica que permita compreender mais a fundo os seus detalhes teóricos, o que será de extrema importância para conseguir reproduzi-las eficazmente em hardware.

Além disso, é necessário identificar a melhor *framework* para desenvolver o modelo conceptual de alto-nível, permitindo assim uma implementação eficiente e comparável com todo o trabalho na área, assim como dos *benchmarks* mais adequados para testar a eficiência da implementação hardware. Por último, convém seleccionar uma aplicação (numa área de reconhecimento de fala, produção musical, entre outras) que permita justificar a pertinência da aplicação prática desta implementação.

## References

- [1] S. Hochreiter and J. Schmidhuber, “Long short-term memory”, English, *Neural Computation*, vol. 9, no. 8, pp. 1735–1735, 1997, ISSN: 08997667.
- [2] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, and J. Schmidhuber, “A novel connectionist system for unconstrained handwriting recognition”, English, *IEEE Trans. Pattern Anal. Mach. Intell. (USA)*, vol. 31, no. 5, pp. 855–68, 2009/05/, ISSN: 0162-8828.
- [3] A. Graves, A.-R. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks”, English, Vancouver, BC, Canada, 2013, pp. 6645–6649.

- 
- [4] A. X. M. Chang, B. Martini, and E. Culurciello, "Recurrent Neural Networks Hardware Implementation on FPGA", 2015/11/17, [Online]. Available: <http://arxiv.org/abs/1511.05552>.
  - [5] H. Siegelmann, "Computation beyond the Turing limit", English, *Science (USA)*, vol. 268, no. 5210, pp. 545–8, 1995/04/28, ISSN: 0036-8075. [Online]. Available: <http://dx.doi.org/10.1126/science.268.5210.545>.
  - [6] J. Cabessa and A. Villa, "Recurrent Neural Networks: A Natural Model of Computation beyond the Turing Limits", English, 2012, pp. 594–9.