Comparação de desempenho do algoritmo de treinamento de uma rede neural em GPU

Marcelo De Felice Lima[1], Wagner M. Nunan Zola[1]

[1]Departamento de Informática – Universidade Federal do Paraná (UFPR)

Curitiba – PR – Brazil

{mflima,wagner}@inf.ufpr.br

Resumo. *Neste trabalho, faz-se uma avaliação de desempenho de uma implementação em CUDA C do algoritmo de treinamento de uma rede neural. Obteve-se speedup de 3x em GPU comparado com o tempo da implementação em Python em CPU através das bibliotecas Keras* e *Tensorflow*.

1. Introdução e descrição do trabalho

Redes neurais são algoritmos de aprendizado de máquina que realizam operações simples de adição e multiplicação para simular a forma como uma rede de neurônios biológicos processa informações, a qual é um sistema complexo, não linear e paralelo [Haykin 2001]. O treinamento do algoritmo de uma rede neural consiste em ajustar os pesos de adição e multiplicação de cada neurônio artificial. Esse processo de treinamento pode ser demorado, por isso buscam-se métodos de otimização. Uma forma de diminuir o tempo de treinamento de uma rede consiste na utilização de GPU para realizar o cálculo de treinamento do algoritmo.

Neste trabalho foi utilizada uma rede do tipo C*onvolutional Neural Network* (CNN) para classificar dígitos manuscritos de uma base de dados conhecida como "MNIST", dividida em base de treinamento e base de teste. A CNN proposta possui duas camadas de convolução, uma camada *flatten* e por último uma camada *dense* totalmente conectada. A primeira camada de convolução recebe imagens da base MNIST em tamanho 28 x 28 pixels, a qual passa por 6 filtros de convolução de tamanho 5 x 5, gerando uma saída com 6 imagens em tamanho 24 x 24. A segunda camada possui um filtro 4 x 4, que percorre as imagens em passos de 4 pixels, reduzindo as imagens para a resolução 6 x 6. A camada *flatten* transforma o conjunto das 6 imagens 6 x 6 em um único vetor de apenas uma dimensão. A última camada, totalmente conectada, gera uma saída com um vetor de 10 argumentos, os quais serão utilizados na classificação das imagens.

2. Resultados, discussão e Conclusão

O treinamento da rede proposta foi feito passando a base de treinamento da MNIST pela rede neural 40 vezes, denominadas épocas. Esse valor foi escolhido ao avaliar a acurácia obtida após determinada quantidade de épocas de treinamento, onde foi observado que a acurácia máxima era obtida após cerca de 40 épocas de treinamento. Foi, então, medido o tempo de execução do algoritmo de treinamento ao final de cada época e realizado um teste de verificação da acurácia utilizando-se a base de teste. O treinamento da rede e classificação da base de dados foram feitos separadamente em GPU e CPU, utilizando a plataforma do *Google Colab*. O código para GPU foi feito em linguagem CUDA C [Kirk e Hwu 2017] e processado em uma GPU Nvidia Tesla T4, arquitetura Turing, com

40 MPs, 1024 *threads* por MP e 1024 *threads* por bloco, a qual foi a GPU alocada pela plataforma. A CPU utilizada foi Intel Xeon com 2 núcleos de 2.20GHz. O código que foi executado exclusivamente em CPU foi feito em *Python*, com auxílio das bibliotecas *Keras* e *Tensorflow*.

A Figura 1(a) apresenta o tempo total decorrido ao término de cada época, para GPU e CPU, e a Figura 1(b) o tempo gasto no treinamento para atingir cada valor de acurácia na base de teste. Por ele observa-se redução no tempo de treinamento utilizando-se GPU. Enquanto a CPU levou 621 segundos para concluir o treinamento em 40 épocas, a GPU levou apenas 219 segundos na mesma tarefa. A Figura 2 apresenta o *speedup* da GPU em relação a CPU, de onde se nota que a GPU foi cerca de 3 vezes mais rápida para processar o treinamento da rede neural. A diminuição no tempo de treinamento ao se usar a GPU se deve ao fato do processamento do treinamento da rede neural ocorrer em paralelo, com várias partes da imagem sendo processadas ao mesmo tempo, ao passo que o processamento na CPU ocorre de forma sequencial.

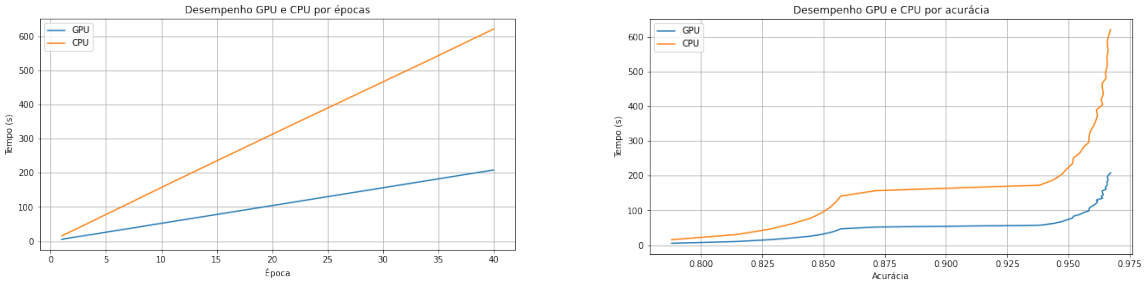

Figura 1. Tempo gasto em GPU e CPU para realizar o treinamento da rede neural. À esquerda, tempo gasto ao final de cada época. À direita o tempo de treinamento necessário para atingir cada valor de acurácia.

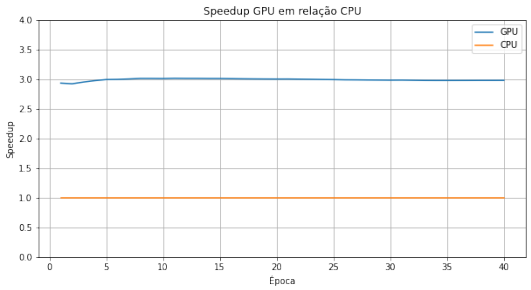

Figura 2. *Speedup* da GPU em relação a CPU para processar o treinamento da rede neural.

Conforme foi verificado, o processamento do treinamento da rede neural em GPU foi cerca de três vezes mais rápido do que em CPU. Nota-se que o uso deste tipo de arquitetura pode trazer redução no tempo de treinamento de redes neurais.

Referências

Haykin, S. S. (2001). Redes Neurais, 2nd ed. BOOKMAN COMPANHIA ED.

Kirk, David B. e Hwu, Wen-mei W. (2017). Programming Massively Parallel Processors: A Hands-on Approach. 3 a ed. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. ISBN : 978-0-12-811986-0.