Aumentando a Velocidade de Treinamento de Rede Neural Convolucional para Classificação de Imagem com Processamento em GPU

Higor de Oliveira Chaves
Universidade Federal
do Mato Grosso do Sul
Coxim, MS 79400–000
Telefone: (67) 99896-4797
Email: higor.chaves@aluno.ufms.br

Rafael Gonçalves de Oliveira Viana
Universidade Federal
do Mato Grosso do Sul
Coxim, MS 79400–000
Telefone: (67) 99950-7979
Email: rafael.viana@aluno.ufms.br

Ramon da Silva V. dos Santos
Universidade Federal
do Mato Grosso do Sul
Coxim, MS 79400–000
Telefone: (67) 98114–5649
Email: ramon.santos@aluno.ufms.br

***Resumo*—Existe atualmente um grande volume de dados sendo transmitidos pelos mais diversos dispositivos disseminados na sociedade mundial. A tendência do mundo moderno, é que esses dados tornem-se um recurso natural, que como qualquer outro, necessita ser refinado para que possa ser bem utilizado. Para que essa grande massa de dados se torne informação útil e relevante porém, algumas técnicas precisam ser aplicadas, técnicas estas que necessitam de um grande poder computacional. Neste artigo, será demonstrado a diferença de perfomance entre CPU e GPU em algoritmos que necessitam de muito processamento. Para tal objetivo foram utilizadas as bibliotecas TensorFlow e cuDDN juntamente com a plataforma CUDA.**

I. Introdução

Com a chegada do conceito de *BigData*, a rede mundial de computadores vem se tornando uma grande mina de ouro. O ouro neste caso seria a informação já processada, porém assim como as minas são mineradas, os dados devem ser minerados.

Os dados são tratados utilizando técnicas (agrupamento, filtro, classificação entre outras), com objetivo de obter informações valiosas, como por exemplo a probabilidade de um produto ser vendido em determinada região, ou ainda a predição de ações na bolsas de valores.

A mineração é apenas uma das etapas de um grande e complexo sistema de análise e refinamento de dados, como a estrutura do KDD. Assim como na mineração de recursos naturais, a mineração de dados necessita de ferramentas e técnicas robustas para poder realizar um maior e melhor processamento. Estas técnicas no entanto, demandam muito poder de máquina e por conta disso são extremamente custosas.

Afim de amenizar a lentidão causada por algoritmos que necessitam de muito recurso computacional, será apresentado neste artigo como a GPU pode se tornar um grande auxiliar no processamento massivo de dados, paralelizando as tarefas e tornando assim o processo como um todo mais veloz.

II. Fundamentação Teórica

A. KDD

Knowledge Discovery in Database ou Descoberta de Conhecimento em Bando de Dados, tem uma função muito importante na produção de conhecimento neste artigo, pois a extração de conhecimento que é usada por esta técnica, não é simplesmente pegar dados, mas sim agrupar informações que passam desapercebidas, porém que tenham relevância e ligação entre si.

B. CUDA

CUDA é uma plataforma de computação paralela inventada pela NVIDIA. Ela é capaz de aumentar significativamente a performance computacional ao utilizar e aproveitar a potência da unidade de processamento gráfico GPU.

Além de trabalhos de artistas e desenvolvedores de jogos, trabalhos inovadores com a tecnologia começaram a sugir. Nascia o movimento da GPU de Propósito Geral (GPGPU).

O CUDA Toolkit conta com um compilador, bibliotecas de matemáticas e ferramentas para depuração e otimização da performance de seus aplicativos, para ficar melhor ele é distribuído gratuitamente, além de sua documentação e manutenção fornecidos pela NVIDIA.

C. cuDDN

É uma biblioteca para o NVIDIA CUDA, para rede neurais profunda que fornece implementações altamente sintonizadas para rotinas padrão, como convolução para trás (propagação), agrupamento, normalização e camadas de ativação. Diversos softwares de redes profundas dependem do cuDDN (Caffe2, MatLab, Microsofot Cognitive Toolkit, TensorFlow, Theano, PyTorch) para aceleracção GPU de alto desempenho, permitindo se concentrar no treinamento de rede neurais e no desenvolvimento de aplicações em vez de gastar tempo no ajuste de desempenho de GPU de baixo nível.

D. TensorFlow

III. Recurso Computacional

Para a realização dos testes será utilizado um computador com as seguintes configurações demostrada na Tabela I.

Tabela I
MY CAPTION

Hardware	Descrição
SO	Debian 9,Stretch Linux Kernel 4.9
CPU	Intel i5-7200U (3.1 GHz, Cache de 3 MB)
GPU	NVIDIA® GeForce® 940MX de 4GB GDDR5
RAM	8GB, DDR4, 2400MHz

IV. ANÁLISE DE TÉCNICAS

Existem cálculos computacionais como os utilizados na engenharia, mídia digital e aplicações científicas que exigem grande poder de processamento, onde a CPU não consegue um bom desempenho.

As GPUs atuam como um co-processador e podem acelerar as aplicações devido ao seu poder de processamento massivamente paralelo em relação ao design dos vários núcleos das CPUs.

Nesta sessão serão apresentadas algunas análises desses cálculos.

1) Multiplicação de Matrizes: Esse é um exemplo trivial da computação pelo seu grande gasto computacional, um exemplo prático séria algumas redes neurais convolucional de classificação de imagem, que utiliza o produto das matrizes para criação de filtros.

Um exemplo prático pode ser observado na Figura **??**, onde foram apresentados três testes de 150^2, 1500^2, 15000^2, todos com valores internos randômicos, a GPU somente foi superior no teste quando o nível de complexidade relativamente grande.

Tabela II
MY CAPTION

Hardware	Descrição
150 X 150	0:00.137
1500 X 1500	0:00.318
15000 X 15000	2:36.198

CONCLUSÃO

O processamento paralelo em GPU tem grande difirença quando os dados computados são de grande complexidade, o que torna essa tarefa muito custosa para a CPU, tornando assim a GPU superior nesses cálculos, porém para cálculos computacionais de baixa complexidade a CPU, se mostrou superior no desempenho. atenção

REFERÊNCIAS

[1] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.