

UM PANORAMA SOBRE TREINAMENTO DE REDES NEURAIIS CONVOLUCIONAIS ENTRE DIFERENTES UNIDADES DE PROCESSAMENTO

Matheus Collares Machado¹

Sandro Da Silva Camargo²

Resumo:

Sistemas especializados que trabalham com machine learning podem aprender com uma grande base de dados e prever futuros resultados com uma margem de erro extremamente baixa. Neste contexto é questionável o quanto a capacidade de processamento de uma CPU (Central Processing Unit) pode influenciar nos tempos de treinamento e teste da rede neural, levando em consideração suas prioridades para com os periféricos e com o sistema operacional. Objetiva-se então uma comparação nos tempos de execuções total (treino e teste), do algoritmo Basenji, em uma CPU e um ambiente de alto desempenho (GPU - Graphics Processing Unit). O algoritmo escolhido para o teste é o Basenji que consiste em uma rede neural convolucional baseada em deep learning que foi executado em uma CPU Intel Xeon E5620 dispondo de 8 gb de RAM e 14 gb de SWAP rodando Ubuntu 16.04 sobre um ambiente virtual de python 3.5.2 e utilizando TensorFlow versão 1.5. O Basenji utiliza dados de genomas em formato FASTA (Se trata de arquivos de texto que contém longas sequências formadas por combinações de letras, correspondentes às iniciais de cada nucleótido) pré alinhados pelo algoritmo chamado Bowtie2. A base de dados escolhida é referente a genomas regulatórios relacionados a biologia do coração e o objetivo da rede neural é aprender com a base de dados e prever novas interações referentes a atividade regulatória sequencial entre cromossomos utilizando uma rede neural profunda convolucional. Considerando o fato de o algoritmo funcionar com sequências enormes de dados genômicos uma rotina de pré - processamento foi necessária para que pudéssemos mapear com precisão as regiões do genoma que podiam ser mapeadas e pular onde o genoma não podia ser mapeado de acordo com o arquivo unmap_macro.bed presente dentro do diretório de instalação do algoritmo. Este arquivo foi disponibilizado pelos desenvolvedores do algoritmo. Após a rotina de pré - processamento a rotina de treinamento foi chamada e até a data de escrita deste artigo ainda está em execução. Espera-se que o desempenho em termos de tempo de execução da CPU seja muito inferior ao tempo de execução na GPU e com isso chegar a conclusão da inviabilidade de utilização de uma CPU para treinamento de redes neurais convolucionais complexas.

Palavras-chave: inteligência artificial, redes neurais, basenji

Modalidade de Participação: Iniciação Científica

**UM PANORAMA SOBRE TREINAMENTO DE REDES NEURAIS CONVOLUCIONAIS ENTRE
DIFERENTES UNIDADES DE PROCESSAMENTO**

¹ Aluno de graduação. mthsmchd5@gmail.com. Autor principal

² Docente. camargo.sandro@gmail.com. Orientador

Anais do 10º SALÃO INTERNACIONAL DE ENSINO, PESQUISA E EXTENSÃO - SIEPE
Universidade Federal do Pampa | Santana do Livramento, 6 a 8 de novembro de 2018

UM PANORAMA SOBRE TREINAMENTO DE REDES NEURAIS CONVOLUCIONAIS ENTRE DIFERENTES UNIDADES DE PROCESSAMENTO

Sistemas especializados que trabalham com *machine learning* podem aprender com uma grande base de dados e prever futuros resultados com uma margem de erro extremamente baixa. Neste contexto é questionável o quanto a capacidade de processamento de uma CPU (*Central Processing Unit*) pode influenciar nos tempos de treinamento e teste da rede neural, levando em consideração suas prioridades para com os periféricos e com o sistema operacional. Objetiva-se então uma comparação nos tempos de execuções total (treino e teste), do algoritmo Basenji, em uma CPU e um ambiente de alto desempenho (GPU - *Graphics Processing Unit*).

O algoritmo escolhido para o teste é o Basenji que consiste em uma rede neural convolucional baseada em deep learning que foi executado em uma CPU Intel Xeon E5620 dispondo de 8 gb de RAM e 14 gb de SWAP rodando Ubuntu 16.04 sobre um ambiente virtual de python 3.5.2 e utilizando TensorFlow versão 1.5.

O Basenji utiliza dados de genomas em formato FASTA (Se trata de arquivos de texto que contém longas sequências formadas por combinações de letras, correspondentes às iniciais de cada nucleótido) pré alinhados pelo algoritmo chamado Bowtie2. A base de dados escolhida é referente a genomas regulatórios relacionados a biologia do coração e o objetivo da rede neural é aprender com a base de dados e prever novas interações referentes a atividade regulatória sequencial entre cromossomos utilizando uma rede neural profunda convolucional.

Considerando o fato de o algoritmo funcionar com sequências enormes de dados genômicos uma rotina de pré - processamento foi necessária para que pudéssemos mapear com precisão as regiões do genoma que podiam ser mapeadas e pular onde o genoma não podia ser mapeado de acordo com o arquivo unmap_macro.bed presente dentro do diretório de instalação do algoritmo. Este arquivo foi disponibilizado pelos desenvolvedores do algoritmo.

Após a rotina de pré - processamento a rotina de treinamento foi chamada e até a data de escrita deste artigo ainda está em execução. Espera-se que o desempenho em termos de tempo de execução da CPU seja muito inferior ao tempo de execução na GPU e com isso chegar a conclusão da inviabilidade de utilização de uma CPU para treinamento de redes neurais convolucionais complexas.