Proposta de tema
MAC0499 - Trabalho de Formatura Supervisionado

Aluno: Marcelo Nascimento Dos Santos Junior

Orientador: Alfredo Goldman vel Lejbman
Orientador: Pedro Henrique Rocha Bruel

Resumo

Neste projeto pretende-se explorar técnicas de aprendizado de máquina voltadas à computação de alto desempenho com foco na paralelização automática de trechos de códigos na linguagem C. A biblioteca escolhida para esta tarefa foi o OpenMP e a configuração de cada trecho será inferida por meio de aprendizado de máquina. Diversos exemplos serão coletados para a fase de treinamento e ao final serão realizadas análises de desempenho dos resultados.

1 Introdução

Aprendizado de máquina (em inglês, machine learning ou ML) é um subgrupo de inteligência artificial que tem o objetivo de obter padrões com o auxílio de algum algoritmo que interprete os dados de uma aplicação. O foco é permitir que uma máquina consiga inferir um resultado futuro sem que esse estado tenha sido efetivamente programado.

O algoritmo recebe uma lista de exemplos de treinamento de casos anteriores, onde cada caso é composto por um conjunto de dados que geram seu respectivo resultado e ao analisar cada elemento da lista, o algoritmo tenta encontrar um padrão que será aplicado na inferência de um conjunto de dados novo que não está presente na lista de treinamento.

Diversos problemas podem ser formulados para que sejam resolvidos por aprendizado de máquina, uma dessas aplicações é o aprendizado aplicado ao código fonte. Bastante presente em ambientes de desenvolvimento integrado (em inglês, Integrated Development Environment ou IDE), esses algoritmos sugerem, geraram e até revisaram códigos com a finalidade de facilitar o processo de desenvolvimento de software.

A programação paralela é um dos processos no desenvolvimento de software que consiste na divisão de uma determinada aplicação em diferentes partes a fim de serem executadas simultaneamente em vários elementos de processamento. Uma das ferramenta utilizadas neste processo é o OpenMP, uma biblioteca baseada em diretivas de compilação que consistem em comentários com sintaxe especial para informar ao compilador como determinado trecho de código deve ser executado em paralelo, o que torna o procedimento mais simples do que outras soluções como pthreads ou MPI.

As diretivas possuem cláusulas que funcionam como parâmetros de uma região do código que se queria paralelizar, com poucas cláusulas é possível paralelizar um grande número de regiões, porém, muitas vezes, essa tarefa se torna inviável com o crescimento das linhas de

código em um projeto, visto que a configuração de uma diretiva, apesar de simples, necessita de conhecimento da lógica do que é executado naquela região.

Iniciativas como as descritas nos artigos [1] e [2] se propõem a contornar esse problema ao utilizar aprendizado de máquina para inferir as diretivas necessárias para paralelizar trechos de código. Os artigos têm abordagens diferentes, porém, nos dois casos, é adaptado um modelo de outro propósito e utilizam-o com uma outra base de treinamento, neste caso, vários exemplos de configurações de diretivas para trechos de loops for da linguagem C.

2 Motivação

Com o avanço da engenharia de software, as soluções implementadas estão cada vez mais complexas e demandam cada vez mais linhas de código, otimizar qualquer tarefa em um ambiente com este pode ser extremamente custoso, visto a necessidade do entendimento sobre o que cada trecho faz. O OpenMP é uma ferramenta prática para paralelizar trechos de código e suas diretivas são facilmente configuráveis. O intuito deste projeto é unir essas duas componentes com o auxílio de aprendizado de máquina.

3 Objetivos

A proposta deste projeto é reproduzir os resultados obtidos no artigo "A machine learning method to variable classification in OpenMP" [1], nele é adaptado o modelo DeepTyper e treinado com exemplos obtidos do benchmark NPB e do framework ROSE. No artigo "Learning to Parallelize in a Shared-Memory Environment with Transformers" é disponibilizada uma base de exemplos extraídos de repositórios do Github e também será utilizada neste projeto para comparação.

4 Metodologia

O DeepTyper é um modelo de inferência de tipos de variáveis do Javascript, é construído sob o framework de Deep Learning CNTK da Microsoft e escrito em Python. O script desmembra o código em tokens e cria um dataset onde cada variável é associada ao seu tipo antes de ser enviado ao modelo construído no CNTK.

Inferir uma diretiva à um loop for é um subespaço do problema que o DeepTyper se propõe a resolver, visto que o mecanismo de associar variáveis à tipos é o mesmo, porém, neste novo problema, é necessário associar as variáveis de um loop for às poucas cláusulas existentes no OpenMP.

O script do DeepTyper é aberto e facilmente editável, o desafio deste projeto consiste em entender a lógica do algoritmo, como ele se relaciona com o CNTK e adaptá-lo ao novo problema.

	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez
1	x	x	x						
2	x	x	x						
3				x	x	x			
4							x	x	

5 Cronograma

1. Obter e tratar as bases de dados de treinamento: Até o momento, serão utilizados os dados do artigo em [2], os gerados pelo software ROSE e do benchmark NPB.

2. Configurar o ambiente de desenvolvimento: Os frameworks CNTK e ROSE necessitam de uma imagem Docker para operar e a tokenização será feita pelo compilador Clang, ou seja, não é possível integrar essas tarefas como em outros frameworks de Deep Learning. Espera-se que uma inferência simples seja feita ao final do período.

3. Realizar experimentos: Treinar os modelos com diversos parâmetros, bases de dados e escopo de variáveis possíveis.

4. Análise dos resultados e escrita da monografia

Referências

1. Yuanyuan Shen, Manman Peng, Qiang Wu, Renfa Li, A machine learning method to variable classification in OpenMP, Future Generation Computer Systems, Volume 140, 2023, Pages 67-78,https://doi.org/10.1016/j.future.2022.10.010.

2. Harel, Re'Em & Pinter, Yuval & Oren, Gal. (2022). Learning to Parallelize in a Shared-Memory Environment with Transformers.