Imperial College London

Department of Mathematics

Solving the Collatz conjecture

Author: Ilia Sobakinskikh (CID: 00000000)

A thesis submitted for the degree of

MSc in Mathematics and Finance, 2022-2023

Declaration

The work contained in this thesis is my own work unless otherwise stated.

Acknowledgements

This is where you usually thank people who have provided useful assistance, feedback,...., during your project.

Abstract

The abstract is a short summary of the thesis' contents. It should be about half a page long and be accessible by someone not familiar with the project. The goal of the abstract is also to tease the reader and make him want to read the whole thesis.

Contents

List of Figures

List of Tables

Introduction

In this thesis, we explore how the inference time of a Transformer Neural Network can be efficiently optimized with applications to real-time anomaly detection in financial time series. The financial time series are price series such as asset prices. Unfortunately, the data is often with errors or outliers that make the downstream data processing tasks useless, unstable or even harmful [1] [2]. Moreover, the amount of financial time-series data has been significantly increasing [3]. Hence, there is a need for better data-cleaning methods in terms of accuracy and in terms of processing speed.

Transformers as a neural network architecture have achieved superior performances in many tasks such as Natural Language Processing and Computer Vision [4]. Time series modelling and especially anomaly detection tasks can benefit from the features of transformers architecture in multiple ways, including the capacity to capture long-range dependencies and interactions [5].

Increasingly powerful hardware, such as field-programmable gate arrays (FPGAs), have seen increasing usage in recent years due to their reconfigurability and high performance [6].

We explore different Transformer architectures for time series modelling and how they can be efficiently implemented on an FPGA board (PYNQ-Z2). In particular, we examine the application of Transformers to detect anomalies in time series and we show how they can be efficiently implemented on an FPGA board to minimize latency or to maximize throughput.

Chapter 1

Methodology

In this chapter, we will describe the main concepts and ideas used in this work.

1.1 Problem Formulation

We consider a multivariate time-series, which is a timestamped sequence of observations/datapoints.

The **Anomaly Detection** task: given a training input time-series T, for any unseen test time-series $\hat{T}$ of length n, we need to predict $Y = \{y_1, ..., y_n\}$, where we use $y_t \in \{0, 1\}$ to denote whether the datapoint at the t-th timestamp of the test set is anomalous (1 denotes an anomalous datapoint).

1.1.1 Transformers

Vanilla Attention layer

1.2 FPGA design

In this section, the main design principles of programming an FPGA board will be described. Readers will be introduced to the common optimization techniques and how they are achieved. The FPGA programming will be done using C++ HLS which is converted to verilog code.

1.2.1 Introduction to FPGA

The progress of hardware acceleration devices like field-programmable gate arrays (FPGAs) enables the achievement of high component density and low power consumption, all the while minimizing latency [6]. They are commonly used to accelerate high-performance, computationally intensive systems (for example, data centers) or to minimize the latency of execution (for example, in high-frequency trading).

1.2.2 FPGA development and HLS

Common Terms

Simulation, Cosimulation

A way to design and debug the solution without running it on the board.

HLS synthesis

In this section, HLS synthesis will be described [7]. It is now the common workflow in the FPGA development because it significantly improves the productivity when working with design.

1.2.3 Common optimizations

In this section, common optimization techniques and how they are achieved will be introduced.

Pipelining

Example code:

```
void toplevel(din_t* a, din_t* b, dout_t* c, int len) {
  vadd: for(int i = 0; i < len; i++) {
#pragma HLS PIPELINE
     c[i] = a[i] + b[i];
  }
}
```

Loop Unrolling

Arrays

TODO: Partitioning

Streams

TODO:

Chapter 2

Experiments

2.1 Datasets

In this section, the datasets used for evaluation will be described.

2.1.1 Numenta Anomaly Benchmark (NAB)

To assess the accuracy of predictions, we use the Numenta Anomaly Benchmark [8] dataset, which contains various real-world time series of temperature sensor readings, CPU utilization of cloud machines, service request latencies, and taxi demands in New York City. It is commonly used to assess the performance of anomaly detection models on time-series data.

2.1.2 FI2010

In [9], authors described the first publicly available benchmark dataset of high-frequency limit order markets for mid-price prediction. The dataset contains 10-day limit order book data from June 2010 for five stocks that are listed on the Helsinki exchange. Each entry in the time series provides price details and aggregate order sizes for the top ten levels on both the bid and offer sides of the market, totaling forty data points. The time series consists of approximately four million messages, representing incoming buy/sell orders or cancellations. the dataset features order book snapshots taken after every 10 messages, resulting in approximately 400,000 records for the five stocks.

A number of versions of the dataset are available using different normalization schemes. We used the not normalized version of the dataset.

For the purpose of this work, we only extract only the mid price from the dataset which will be used for anomaly detection task.

Synthetic outliers

Since the dataset is not labeled, we have to inject synthetic anomalies into the dataset. We employ the approach similar to [10] with a slight modification. The algorithm can be summarized as follows:

1. Select n samples from the time series which will be contaminated (i.e., anomalous)
2. Replace the sample S_i with $\hat{S}_i = S_i(1+\delta)$ where δ is the injected outlier in the return space.

Authors model δ as a uniformly distributed random variable $\mathcal{U}[0, \rho]$. We instead use the normal distribution with matching mean and standard deviation of the returns time series.

2.2 Accuracy

2.3 Performance/Speed

2.4 Resource utilization on FPGA

Conclusion

2.5 Future work

Bigger FPGA boards.

Evaluation of performance on more recent financial market data.

Appendix A

Code

A.1 Efficient matrix multiplication

This is Appendix A.1, which usually contained supporting material, or complicated proofs that might make the main text above less readable / fluid.

Bibliography

[1] Thomas Neil Falkenberry CFA. High frequency data filtering, Sep 2008.

[2] Owen Vallis, Jordan Hochenbaum, and Twitter. Introducing practical and robust anomaly detection in a time series.

[3] Mohiuddin Ahmed, Nazim Choudhury, and Shahadat Uddin. Anomaly detection on big data in financial markets. In *2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 998–1001, 2017.

[4] Anthony Gillioz, Jacky Casas, Elena Mugellini, and Omar Abou Khaled. Overview of the transformer-based models for nlp tasks. In *2020 15th Conference on Computer Science and Information Systems (FedCSIS)*, pages 179–183, 2020.

[5] Qingsong Wen, Tian Zhou, Chaoli Zhang, Weiqi Chen, Ziqing Ma, Junchi Yan, and Liang Sun. Transformers in time series: A survey. 2022.

[6] Andreea-Ingrid Funie, Liucheng Guo, Xinyu Niu, Wayne Luk, and Mark Salmon. Custom framework for run-time trading strategies. In Stephan Wong, Antonio Carlos Beck, Koen Bertels, and Luigi Carro, editors, *Applied Reconfigurable Computing*, pages 154–167, Cham, 2017. Springer International Publishing.

[7] Xilixn Inc. Vitis High-Level Synthesis User Guide, may 10 2023. [Online; accessed 2023-07-16].

[8] Subutai Ahmad, Alexander Lavin, Scott Purdy, and Zuha Agha. Unsupervised real-time anomaly detection for streaming data. *Neurocomputing*, 262:134–147, 11 2017. [Online; accessed 2023-07-19].

[9] Adamantios Ntakaris, Martin Magris, Juho Kanniainen, Moncef Gabbouj, and Alexandros Iosifidis. Benchmark dataset for mid-price forecasting of limit order book data with machine learning methods. 2017.

[10] Stéphane Crépey, Noureddine Lehdili, Nisrine Madhar, and Maud Thomas. Anomaly Detection in Financial Time Series by Principal Component Analysis and Neural Networks. *Algorithms*, 15(10):385, oct 19 2022. [Online; accessed 2023-07-19].