VIETNAM GENERAL CONFEDERATION OF LABOR

**TON DUC THANG UNIVERSITY**

**INFORMATION TECHNOLOGY FACULTY**



**FINAL PROJECT OF INTRODUCTION TO MACHINE LEARNING COURSE**

**Report of comparison in Optimizer methods and Continual Learning and Test Production in Machine Learning**

*Teacher*: **PhD LE ANH CUONG**

*Student*: **NGUYEN KHAC HUY – 521H0502**

Class **: 21H50301**

Course  **: 25**

**HO CHI MINH, NĂM 2023**

VIETNAM GENERAL CONFEDERATION OF LABOR

**TON DUC THANG UNIVERSITY**

**INFORMATION TECHNOLOGY FACULTY**



**FINAL PROJECT OF INTRODUCTION TO MACHINE LEARNING COURSE**

**Report of comparison in Optimizer methods and Continual Learning and Test Production in Machine Learning**

Teacher: **PhD LE ANH CUONG**

Student: **NGUYEN KHAC HUY**

Class **: 21H50301**

Course  **: 25**

**HO CHI MINH, NĂM 2023**

ACKNOWLEDGEMENT

To complete this final project, I would like to express our gratitude to Ton Duc Thang University for providing the necessary facilities and to the teachers who have supported us throughout our studies at the university. Moreover, I love to give thanks to our parents who gave us chances to approach knowledge at TDTU and motivated, supported us a lot.

Especially, we would like to thank Mr. Le Anh Cuong for teaching us with great dedication and detail, so that I have enough knowledge to use for this essay. Due to my limited experience and knowledge, I am sure that there are some mistakes in my work. I sincerely hope to receive feedbacks and constructive criticism from my teachers so that I can complete this essay more effectively.

I would like to express my heartfelt thanks and wish teacher good health.

**THE PROJECT WAS COMPLETED**

**AT TON DUC THANG UNIVERSITY**

I hereby declare that this is my/our own project product and is guided by Dr. Nguyen Van A;. The research content and results in this topic are honest and have not been published in any form before. The data in the tables for analysis, comments, and evaluation were collected by the author from different sources and clearly stated in the reference section.

In addition, the project also uses a number of comments, assessments as well as data from other authors and other organizations, all with citations and source notes.

**If any fraud is discovered, I will take full responsibility for the content of my project**. Ton Duc Thang University is not involved in copyright violations caused by me during the implementation process (if any).

*Ho Chi Minh, 12th December 20, 2023*

*Author*

*(sign and write full name)*

*Nguyen Khac Huy*

CONFIRMATION AND EVALUATION OF LECTURERS

**Confirmation of instructors**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ho Chi Minh, day month year

(sign and write full name)

**Evaluation by instructors marking report**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Ho Chi Minh, day month year

(sign and write full name)

SUMMARY

This report discusses about comparison of difference optimizer methods. In chapter two of this report I will discuss about Continual Learning and Test Production for solving practical problem.

TABLE OF CONTENTS

[LỜI CẢM ƠN i](#_Toc387692905)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN i](#_Toc387692906)

[TÓM TẮT i](#_Toc387692907)

[MỤC LỤC 1](#_Toc387692908)

[DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ 1](#_Toc387692909)

[CHƯƠNG 1 – MỞ ĐẦU 1](#_Toc387692910)

[1.1 Tiểu mục cấp 1 1](#_Toc387692911)

[1.1.1 Tiểu mục cấp 2 1](#_Toc387692912)

[1.1.1.1 Tiểu mục cấp 3 1](#_Toc387692913)

[1.1.1.2 Tiểu mục cấp 3 tiếp theo. 1](#_Toc387692914)

[1.1.2 Tiểu mục cấp 2 tiếp theo 1](#_Toc387692915)

[1.2 Nội dung của chương này 1](#_Toc387692916)

[CHƯƠNG 2 – TỔNG QUAN 1](#_Toc387692917)

[1.1 Trình bày công thức toán học 1](#_Toc387692918)

[1.2 Trình bày một hình vẽ, sơ đồ 1](#_Toc387692919)

[CHƯƠNG 3 – CƠ SỞ LÝ THUYẾT / NGHIÊN CỨU THỰC NGHIỆM 1](#_Toc387692920)

[3.1 Chèn bảng: 1](#_Toc387692921)

[3.2 Viết tắt 1](#_Toc387692922)

[3.3 Trích dẫn 1](#_Toc387692923)

[3.3.1 Tài liệu tham khảo và cách trích dẫn 1](#_Toc387692924)

[3.3.2 Qui định của Khoa Công nghệ thông tin 1](#_Toc387692925)

CHAPTER 1 – LEARNING AND COMPARE OPTIMIZER METHODS IN TRAINING MODEL.

1.1 Learning about Optimizer methods.

1.1.1 Definition of optimizer method

An optimizer method in machine learning is a critical component of the training process for many machine learning models, particularly deep learning models. Its primary function is to minimize a predefined loss or cost function by iteratively adjusting the model's parameters (weights and biases) during training. The goal is to find the optimal set of parameters that results in the model making accurate predictions on a given task.

1.1.2 Key characteristics of optimizer method

Loss Function: The optimizer relies on a loss function, which quantifies the error or difference between the model's predictions and the actual target values. The goal is to minimize this loss.

Learning Rate: The learning rate is a hyperparameter that determines the step size or the rate at which the optimizer updates the model's parameters. Setting the appropriate learning rate is crucial, as too large a rate can lead to overshooting the optimal solution, while too small a rate can result in slow convergence.

Optimizers play a crucial role in training neural networks and other machine learning models. The choice of optimizer and its hyperparameters can significantly impact the training process, affecting training speed, convergence, and the quality of the learned model.

1.1.3 Optimizer methods

1.1.3.1 Gradient Descent (GD)

Gradient Descent is an optimization method used to minimize a loss function by iteratively adjusting model parameters in the direction of the steepest decrease in the gradient. It computes the gradient using the entire training dataset in each iteration.

Pros:

* Conceptual Simplicity: GD is easy to understand and implement.
* Global Convergence: For convex loss functions, GD is guaranteed to converge to the global minimum.

Cons:

* Slow Convergence: It can be slow for large datasets since it computes gradients for the entire dataset in each iteration.
* Local Minima: It may get stuck in local minima for non-convex loss functions.

1.1.3.2 Stochastic Gradient Descent (SGD)

Stochastic Gradient Descent is a variation of GD where model parameters are updated using the gradient computed for a single randomly chosen data point (or mini-batch) in each iteration.

Pros:

* Faster Convergence: SGD often converges faster than GD, especially for large datasets, due to the reduced computational cost per iteration.
* Escape Local Minima: The stochasticity in updates helps escape local minima.

Cons:

* High Variance: The high variance in updates can lead to oscillations during training.
* Tuning: It may require careful tuning of the learning rate and learning rate schedules.

1.1.3.3 Mini-Batch Gradient Descent

Mini-Batch Gradient Descent is a compromise between GD and SGD. It updates model parameters using a small random subset (mini-batch) of the training data in each iteration.

Pros:

* Balance: It strikes a balance between the robustness of GD and the speed of SGD, making it suitable for most cases.
* Parallelism: Mini-batch processing can be efficiently parallelized on modern hardware.

Cons:

* Hyperparameter Sensitivity: The choice of mini-batch size can be sensitive and may require tuning.
* Learning Rate: It can still be sensitive to learning rate choice.

1.1.3.4 Momentum

Momentum extends basic gradient descent by introducing a momentum term that accumulates past gradients. It updates parameters with a combination of the current gradient and the accumulated momentum.

Pros:

* Accelerated Convergence: Momentum accelerates convergence by smoothing updates and dampening oscillations.
* Escape Local Minima: It helps escape local minima by maintaining velocity.

Cons:

* Hyperparameter Tuning: Momentum requires tuning of the momentum hyperparameter.
* Sensitivity to Learning Rate: It can be sensitive to the choice of learning rate.

1.1.3.5 Learning Rate Scheduling

Learning Rate Scheduling dynamically adjusts the learning rate during training to control its behavior over time. Common methods include step decay, exponential decay, and adaptive learning rate methods.

Pros:

* Improved Convergence: Learning rate scheduling improves convergence and robustness by adapting the learning rate to the problem.
* Adaptation: It adapts to the specific problem during training.

Cons:

* Complexity: Selecting and tuning the scheduling method can be complex.
* Not Universally Optimal: It may not work optimally for all problems, and experimentation is often required.

CHƯƠNG 2 – CONTINUAL LEARNING AND TEST PRODUCTION FOR SOLVING PROBLEMS.

This chapter demonstrates my research about Continual Learning and Test Production. Moreover, this chapter also shows approach and how to apply Continual Learning, Test production in creating solution to address problems.

2.1 Continual Learning

2.1.1 What is Continual Learning ?

Continual Learning (CL) focuses on developing models to learn new tasks while retaining information from previous tasks. CL is an important area of research as it addresses the real-world scenario where the data and tasks are constantly changing, and a model must adapt to these changes without forgetting previous knowledge.

Unlike traditional machine learning models, which are often trained on fixed datasets and do not adapt well to new information, continual learning models evolve over time.

2.1.2 Challenge in Continual Learning?

- Incomplete and growing dataset:

* In many large-scale learning scenarios, not all training data might be available when we want to be training a network. Even worse, the set of the task may be dynamically grow as new tasks are introduced.
* However, if the classes we had in the early stages of learning significantly differs from the new classes, utilization of prior knowledge may degenerate performance.

- Catastrophic Forgetting: Introduction of new units can also result in semantic drift or catastrophic forgetting, where original meaning of the features change as they fit to later tasks.

2.1.3 Strategies and Techniques in Continual Learning

Strategies: Continual Learning (CL) strategies are methods that help machine learning models learn new things without forgetting what they already know. These strategies include techniques like keeping important information safe, practicing with old data, and changing the model as needed. Imagine it's like learning new dance moves without forgetting the old ones you've mastered. CL strategies make sure the learning process is smooth and that the model keeps getting better over time.

Techniques:

2.1.4 Continual Learning(CL) algorithms

*Elastic Weight Consolidation (EWC):* It's like using a gentle brake when learning new things. EWC prevents old knowledge from fading by penalizing big changes in what the model already knows.

*Synaptic Intelligence (SI):* SI is similar to EWC but adds a smarter penalty. It cares not only about how much things change but also which parts are most important for the model.

*Experience Replay (ER):* Imagine reviewing your old notes before tackling new subjects. ER stores old lessons and revisits them during new learning to avoid forgetting.

*Gradient Episodic Memory (GEM):* GEM is like keeping a record of how you solved problems in the past. It uses this record to make sure you don't mess up the old stuff while learning new things.

*Progressive Neural Networks (PNN):* PNN is like adding extra pages to your notebook for new topics. It expands the model's capacity as it encounters new tasks.

*Learning without Forgetting (LwF):* LwF is like a teaching assistant who helps you remember old lessons when you're learning new ones. It ensures the new knowledge doesn't erase the old.

2.2 Test Production

Test production is the process of creating and implementing tests to evaluate the performance, accuracy, and reliability of your machine learning solution. When developing a machine learning solution to solve a problem, test production plays a critical role:

* Ensuring Quality: Testing ensures that your model works as expected and meets the desired quality standards. It's like quality control in manufacturing.
* Detecting Issues: Testing helps identify problems, errors, or biases in your model's predictions or behavior. It's similar to diagnosing issues in a medical check-up.
* Validating Results: Test production validates that your model's output aligns with your problem-solving objectives. It's like cross-checking your answers in a math test.
* Robustness Assessment: Tests evaluate how well your model performs under various conditions, including challenging or unexpected situations.

There are different types of testing in machine learning:

* Unit Testing: Testing individual components or functions of your code to verify their correctness.
* Integration Testing: Testing how different components or modules of your machine learning system work together.
* Model Evaluation: Evaluating the performance of your machine learning model on real-world data.

Understanding and applying these principles of test production in machine learning, you can ensure that your models are not only accurate but also robust, fair, and reliable in various real-world scenarios.

2.3 Real – world problem

Problem: Personalized Recommendation Systems for E-commerce on Shopee.

Scenario: Shopee is a popular e-commerce platform in Southeast Asia, where millions of users buy a wide range of products daily. The platform aims to enhance user experience by providing personalized product recommendations based on user behavior and preferences

**Data Input**: For a personalized recommendation system in an e-commerce platform like Shopee, the dataset input is essential and consists of various components. This includes user interaction data, such as user IDs, product IDs, timestamps, and interaction types, which capture user behavior. Additionally, user profiles, which can include demographic information, preferences, and historical purchases, may be incorporated if available. Product data provides details such as product descriptions, categories, and features, while catalog information encompasses all available products, including new additions. User feedback data, which includes ratings, reviews,

**Production**: In the production phase, the recommendation system takes shape through a series of steps. Data collected from users' interactions is preprocessed to clean and transform it. Feature engineering is employed to create relevant features for recommendations. Machine learning models are trained and evaluated using metrics like MAP or RMSE. Incremental learning techniques are implemented to adapt to changing user preferences. The system generates personalized product recommendations based on historical interactions and preferences. User feedback is actively incorporated to refine the models.

Apply Test Production:

* A/B Testing: To compare the performance of the existing recommendation model (control group) with an enhanced model or algorithm (test group) in a controlled experiment. This helps assess whether the changes made to recommendations have a positive impact on user engagement and conversions.
* Online Evaluation Metrics: To continuously monitor and evaluate the quality of recommendations in real-time using metrics like online AUC or MAP. This ensures that the recommendation system maintains a high level of performance as user behavior and preferences evolve.
* Reinforcement Learning: To use reinforcement learning to optimize the recommendation policy by training an agent to interact with the recommendation system. This step aims to improve recommendations based on user feedback and maximize user engagement and satisfaction.
* Bandit Algorithms: To balance exploration (trying new recommendations) and exploitation (leveraging known preferences) in recommendation. Bandit algorithms adaptively select and recommend items to users, aiming to provide more relevant recommendations while learning from user interactions.
* Dynamic Pricing and Discounts: To enhance recommendations by offering dynamic pricing and personalized discounts. The goal is to boost sales revenue and conversion rates by tailoring pricing and promotions to individual user preferences and behaviors.
* Continuous Model Training: To keep recommendation models up-to-date with the latest user interactions and preferences. Continuous model training ensures that recommendations remain relevant and accurate over time, leading to improved user engagement and satisfaction.

**TÀI LIỆU THAM KHẢO**

**PHỤ LỤC**