

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN**



BÁO CÁO CUỐI KỲ MÔN NHẬP MÔN HỌC MÁY

***Người hướng dẫn:* PGS.TS LÊ ANH CƯỜNG**

***Người thực hiện:* PHẠM TIẾN SANG – 52000794**

TRẦN THANH NHẬT THIÊN – 52100932

Lớp : 20050301

Khoá : 24

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN**



BÁO CÁO CUỐI KỲ MÔN NHẬP MÔN HỌC MÁY

Người hướng dẫn: PGS.TS LÊ ANH CƯỜNG

Người thực hiện: PHẠM TIẾN SANG – 52000794

TRẦN THANH NHẬT THIÊN – 52100932

Lớp : 20050301

Khoá : 24

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023

LỜI CẢM ƠN

Để hoàn thành bài tiểu luận này, lời đầu tiên, chúng em xin trân trọng cảm ơn trường Đại Học Tôn Đức Thắng vì đã tạo điều kiện về cơ sở vật chất với hệ thống thư viện trực tuyến hiện đại, đa dạng các loại sách, tài liệu thuận lợi cho việc tìm kiếm, nghiên cứu thông tin. Đặc biệt, em xin gửi lời cảm ơn sâu sắc đến giảng viên bộ môn – GV.Lê Anh Cường, trong quá trình học tập môn Nhập môn học máy. Thầy đã giảng dạy tận tình, truyền đạt những kiến thức quý báu cho chúng em trong suốt thời gian học tập vừa qua. Từ những kiến thức này, chúng em đã dần trả lời được những câu hỏi trong bài báo cáo này. Thông qua bài báo cáo này em đã trình bày được những kiến thức quý báu mà em được tiếp thu từ thầy.

Do chưa có nhiều kinh nghiệm làm đề tài cũng như những hạn chế, thiếu sót về kiến thức và khả năng lý luận, trong bài báo cáo chắc chắn sẽ không tránh khỏi những thiếu sót. Em rất mong nhận được sự nhận xét, ý kiến đóng góp, phê bình từ phía Thầy để bài báo nào này được hoàn thiện hơn và rút kinh nghiệm về sau.

Lời cuối cùng, kính chúc thầy thật nhiều sức khỏe và thành công hơn trên con đường giảng dạy gieo mầm kiến thức quý báu cho chúng em trên con đường trao dồi kiến thức ở giảng đường đại học quý báu này.

ĐỒ ÁN ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Tôi xin cam đoan đây là sản phẩm đồ án của riêng tôi / chúng tôi và được sự hướng dẫn của PGS.TS Lê Anh Cường;. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình. Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày 23 tháng 12 năm 2023

Tác giả

(ký tên và ghi rõ họ tên)

Sang

Phạm Tiến Sang

Thiên

Trần Thanh Nhật Thiên

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

Phần xác nhận của GV hướng dẫn

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

Phần đánh giá của GV chấm bài

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

TÓM TẮT

Đây là báo cáo cuối kỳ môn Nhập Môn Học Máy gồm 2 phần. Phần 1: sinh viên tìm hiểu về các phương pháp Optimizer, Continual Learning và Test Product. Phần 2: Sinh viên tiến hành xây dựng 1 model machine learning dựa trên các thuật toán đã được học.

MỤC LỤC

LỜI CẢM ƠN	i
PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN	iii
TÓM TẮT	iv
MỤC LỤC	1
BÀI 1	2

BÀI 1

Thuật toán tối ưu (optimizer) mô hình học máy là một bước quan trọng trong việc huấn luyện mô hình. Chúng là bước đầu để xây dựng nên một mô hình mạng neural dùng để học các feature hay pattern từ dữ liệu đầu vào, tìm weight và bias phù hợp và tối ưu model.

Có nhiều phương pháp optimizer, mỗi phương pháp sẽ ảnh hưởng đến tốc độ và hiệu suất khác nhau. Dưới đây là một số phương pháp optimizer dùng để huấn luyện mô hình học máy:

1 Gradient descent

Khi tìm giá trị nhỏ nhất của 1 hàm số trong bài toán tối ưu, ta thường tìm giá trị đạo hàm của nó bằng 0. Nhưng đối với các bài toán phức tạp hay nhiều biến, việc tìm đạo hàm rất khó, nhiều lúc bất khả thi, từ đó hình thành nên cách thức tìm tiệm cận đến điểm cực tiểu của hàm số trong một vòng lặp. Cách thức này gọi là Gradient descent (giảm dần độ dốc), sử dụng gradient của hàm mất mát

$$\text{Công thức : } x_{t+1} = x_t - \text{learnrate} * \text{gradient}(x_t)$$

Trong đó: x_{t+1} : các tham số của model tại thời gian t+1

x_t : các tham số của model tại thời gian t

learnrate: trọng số tốc độ học, kiểm soát bước đếm giữa các tham số

gradient: gradient của hàm mất mát trong thời gian t

2 Stochastic gradient descent

Là biến thể của gradient descent. Thay vì sau mỗi vòng lặp ta cập nhật weight 1 lần thì nếu có N điểm dữ liệu thì ta sẽ cập nhật trọng số N lần. Biến thể này cập nhật các tham số bằng cách áp dụng gradient với tập con ngẫu nhiên của train data

Công thức: $x_{t+1} = x_t - \text{learnrate} * \text{gradient}(x_t; x_i, y_i)$, với điểm dữ liệu ngẫu nhiên (x_i, y_i)

Trong đó: x_i : điểm dữ liệu ngẫu nhiên

y_i : giá trị tương ứng với điểm dữ liệu ngẫu nhiên

3 Momentum/Gradient descent with momentum

Phương pháp momentum đưa động lượng vào phương trình để tăng tốc độ hội tụ, nhất là trong các trường hợp có gradient thưa hoặc nhiều noise. Phương pháp này khắc phục 2 nhược điểm của gradient descent: learning rate và điểm dữ liệu đầu.

$$\text{Công thức : } v_{t+1} = \text{momentum} * v_t + (1 - \text{momentum}) * \text{gradient}(x_t); x_{t+1} = x_t - \text{learnrate} * \text{gradient}(x_t; x_i, y_i)$$

Trong đó: v_{t+1} : động lượng tại thời gian $t+1$

momentum: hệ số động lượng, kiểm soát gradient

v_t : động lượng tại thời điểm t

4 Adaptive gradient (Adagrad)

Các phương pháp đều có learning rate không đổi trong quá trình train data. Phương pháp adagrad biến learning rate thành 1 tham số và điều chỉnh learning rate cho từng tham số trong data dựa trên các gradient trước sau mỗi thời điểm t

$$\text{Công thức : } G_{t+1} = G_t + (\text{gradient}(\theta_t))^2; \theta_{t+1} = \theta_t - \frac{n}{\sqrt{G_t + \epsilon}} \text{gradient}(\theta_t)$$

Trong đó: G_{t+1} : gradient bình phương tích lũy tại thời gian $t+1$

G_t : ma trận chéo của tổng bình phương gradient

ϵ : hằng số nhỏ để tránh lỗi (trường hợp chia cho 0)

5 Root mean square propagation (RMSprop)

RMSprop giải quyết vấn đề learning rate giảm dần của phương pháp adagrad bằng cách chia learning rate cho căn bậc hai của bình phương các gradient trước

$$\text{Công thức : } E[g^2]_t = \text{decay} * E[g^2]_{t-1} + (1 - \text{decay}) * \text{gradient}(\theta_t)^2; \theta_{t+1} = \theta_t - \frac{\text{learnrate}}{\sqrt{E[g^2]_t + \epsilon}} \text{gradient}(\theta_t)$$

Trong đó: $E[g^2]_t$: Trung bình exponential move của bình phương gradient ở thời gian t

decay: Hệ số decay, thêm vào bình phương gradient hiện tại

6 Adaptive moment estimation (Adam)

Adam là sự kết hợp của momentum và RMSprop, duy trì trung bình các gradient trước và các gradient bình phương

Công thức: $m_{t+1} = momentum_1 * m_t + (1 - momentum_1) * gradient(\theta_1)$; $v_{t+1} = momentum_2 * v_t + (1 - momentum_2) * gradient(\theta_2)$; $m^*_{t+1} = \frac{m_{t+1}}{a - momentum_1^{t+1}}$; $v^*_{t+1} = \frac{v_{t+1}}{1 - momentum_2^{t+1}}$; $\theta_{t+1} = \theta_t - \frac{learnrate}{\sqrt{v^*_{t+1} + \epsilon}} * m^*_{t+1}$

Trong đó: m_{t+1} : Ước tính thời điểm thứ 1 tại thời gian t+1

v_{t+1} : Ước tính thời điểm thứ 2 tại thời gian t+1

m^*_{t+1} : Ước tính thời điểm thứ 1 đã điều chỉnh độ sai lệch tại thời gian t+1

v^*_{t+1} : Ước tính thời điểm thứ 2 đã điều chỉnh độ sai lệch tại thời gian t+1

Qua các phương pháp optimizer trên, ta có thể so sánh chúng với các tiêu chí như sau:

- Tốc độ: Adam > RMSprop > Momentum > SGD > Adagrad > Gradient descent (Adam và RMSprop nhanh nhất do có yếu tố động lượng)
- Xử lý dữ liệu noise: Adam > RMSprop > Momentum > SGD > Gradient Descent > Adagrad (Learning rate thích ứng của Adam và RMSprop khó thay đổi theo gradient noise hơn)
- Ứng dụng vào dữ liệu thưa: Adagrad > RMSprop > Momentum > Adam > SGD > Gradient Descent (Learning rate thích ứng của Adagrad có lợi thế hơn cho các trường hợp có dữ liệu thưa thớt)
- Độ tối ưu trong tính toán: Adam > RMSprop > Momentum > SGD > Adagrad > Gradient Descent (Adam và RMSprop sử dụng learning rate thích ứng nên thường sẽ có ít lần lặp hơn các phương pháp khác)

Dựa trên các tiêu chí đó, ta thấy rằng Adam là phương pháp hoạt động khá tốt, tính toán mức tối thiểu nhanh hơn so với các phương pháp khác.

TÀI LIỆU THAM KHẢO

Tiếng Việt

- [1] T. T. Trục, “Optimizer - Hiểu sau về các thuật toán tối ưu,” 17 10 2020. [Trực tuyến]. Available: <https://shorturl.at/dehyN>. [Đã truy cập 22 12 2023].
- [2] V. H. Tiệp, "Gradient Descent," 12 1 2017. [Online]. Available: <https://shorturl.at/bLU15>. [Accessed 20 12 2023].

Tiếng Anh

- [3] A. Smith, "A complete Introduction to Continual Learning," 05 04 2023. [Online]. Available: <https://shorturl.at/pvMN7>. [Accessed 20 12 2023].

PHỤ LỤC

Phần này bao gồm những nội dung cần thiết nhằm minh họa hoặc hỗ trợ cho nội dung luận văn như số liệu, biểu mẫu, tranh ảnh. . . . nếu sử dụng những câu trả lời cho một *bảng câu hỏi* thì *bảng câu hỏi mẫu* này phải được đưa vào phần *Phụ lục ở dạng nguyên bản* đã dùng để điều tra, thăm dò ý kiến; không được tóm tắt hoặc sửa đổi. Các tính toán mẫu trình bày tóm tắt trong các biểu mẫu cũng cần nêu trong Phụ lục của luận văn. Phụ lục không được dày hơn phần chính của luận văn