ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

ÔN THI
MÔN KỸ THUẬT HỌC SÂU VÀ ỨNG DỤNG

KHOA: KHOA HỌC MÁY TÍNH

HOMEWORK:

GV hướng dẫn:

Nhóm thực hiện:

1. Trương Thành Thắng – 20521907

Sự phát triển của các mạng học nông, học sâu đời đầu trên ảnh, văn bản.

Để giữ vững thế giới machine learning, luôn xảy ra các cuộc chiến tranh giữa các trường phái anh hùng mô hình và các quái vật dữ liệu. Trong cuộc chiến này, các anh hùng mô hình phải giải quyết các bài toán mà các quái vật dữ liệu đặt ra.

Khi đó, có một trường phái tên là Regression Warriors gồm có các anh hùng mạng học nông là linear regression, logistic regression và softmax regression. Họ sử dụng các môn võ tuyến tính để đối đầu với các quái vật dạng bảng. Trong đó linear regression là một nhân vật đầy tự tin, anh luôn tự hào về khả năng dự đoán giá trị liên tục một cách chính xác, tuy nhiên anh không thể giải quyết các bài toán phân lớp. Logistic Regression là một nhân vật trẻ trung và năng động. Anh ta sở hữu khả năng giải quyết các bài toán phân lớp nhị phân một cách chính xác. Còn Softmax Regression là một nhân vật đa năng. Anh ta có thể giải quyết các bài toán phân lớp cho nhiều lớp và sử dụng hàm softmax để tính xác suất một cách chính xác.

Tuy nhiên, trường phái Regression Warriors không hiệu quả khi đối đầu bài toán có kết quả đầu ra phụ thuộc phi tuyến với dữ liệu đầu vào. Đối đầu với đối thủ này thì có nhân vật NN, anh ta là một nhân vật mạnh mẽ, anh ta mở rộng không gian biểu diễn quan hệ dữ liệu bằng cách tăng số lớp biến đổi trung gian và phép biến đổi phi tuyến giữa dữ liệu đầu vào và đầu ra. Tuy nhiên đối với sự đa dạng của quái vật dữ liệu dạng ảnh thì NN tỏ ra thiếu năng lực, khi đó anh ta cần lượng lớn dữ liệu để học và thời gian tính toán rất lâu.

Bởi vậy, đó chính là động lực phát triển và hình thành nên một trường phái khác đó là Convolutional Heroes, trường phái này bao gồm các mạng học sâu như AlexNet, VGG-16, ResNet, DenseNet, MobileNet. Họ sử dụng các môn võ Convolutional Neural Network (CNN) để xử lí các tên quái vật dạng ảnh. Trong đó, AlexNet là một trong những người anh đầu đời, đặc biệt nổi bật bởi việc sử dụng ReLU làm hàm kích hoạt và sử dụng tầng Dropout để tránh overfitting. Tuy nhiên, AlexNet vẫn còn nhiều hạn chế về mặt kích thước và sức mạnh tính toán. VGG-16 là người anh kề, anh ta được đánh giá cao vì việc sử dụng các tầng 3x3 để tăng cường độ sâu của mạng. Tuy nhiên, VGG-16 vẫn còn nhiều hạn chế về mặt tính toán vì sức mạnh tính toán không đủ để hỗ trợ việc huấn luyện mạng. ResNet là người anh nổi bật nhất, anh ta khắc phục được hạn chế của VGG-16 bằng cách sử dụng các tầng skip connection để giảm chiều sâu của mạng và giữ được thông tin quan trọng. Điều này giúp cho ResNet tính toán nhanh hơn cả. Gần giống với ResNet, DenseNet là một nhân vật quyết đoán với tầm nhìn rộng và kiên trì. DenseNet đặt ra mục tiêu giữ gắn sự liên kết giữa các tầng, giúp cho việc truyền thông thông tin giữa các tầng trở nên dễ dàng và hiệu quả hơn. Ưu điểm của DenseNet là cho kết quả chính xác và ít bị overfitting, nhưng nhược điểm là sử dụng nhiều tài nguyên hơn so với các mạng khác.

MobileNet là một nhân vật linh hoạt và cẩn thận. Anh ta hiểu rằng việc sử dụng mạng học sâu trên các thiết bị di động vẫn còn phụ thuộc vào tài nguyên, vì vậy anh ta muốn tạo ra một mạng mà không cần nhiều tài nguyên. MobileNet lấy một cách triết lý "ít là tốt" và sử dụng các tầng Depthwise Separable Convolution và Pointwise Convolution để giảm độ dài của mạng và giảm độ phức tạp tính toán, đồng thời vẫn giữ được độ chính xác cao. Tuy nhiên trường phái Convolutional Heros vẫn có hạn chế, dữ liệu đầu vào và đầu ra của trường phái này ở vector với các thành phần độc lập nhau, vì vậy họ sẽ không phù hợp đối đầu với những quái vật dạng văn bản.

Có một trường phái chuyên đối đầu với quái vật dạng văn bản tên là Sequence Savants, gồm các mạng học sâu như RNN LSTM, Seq2seq, encoder-decoder và một chuyên gia kỹ thuật Attention. Họ sử dụng các môn võ sequence modeling để xử lí quái vật dạng văn bản. Trong đó, RNN (Recurrent Neural Network) là một nhân vật trung bình, nó được sử dụng để giải quyết các bài toán liên quan đến dữ liệu có liên quan giữa các thời điểm. Tuy nhiên, RNN có một nhược điểm là không khả dụng để xử lý các bài toán về dữ liệu dài với mối quan hệ trong tương lai xa với hiện tại. LSTM (Long Short-Term Memory) là một nhân vật mạnh mẽ hơn, nó có khả năng giải quyết được các vấn đề của RNN về dữ liệu dài và mối quan hệ trong tương lai xa với hiện tại. Seq2Seq (Sequence to Sequence) là một nhân vật tương tự như LSTM, anh ta có ưu điểm là tối ưu hóa việc chuyển đổi thông tin giữa các đầu vào và đầu ra khác nhau, nhưng cũng có nhược điểm là khả năng phân tích thông tin vẫn còn hạn chế. Nhân vật Attention là một chuyên gia trong việc tìm kiếm và phân tích thông tin quan trọng từ một bộ dữ liệu lớn. Nó có ưu điểm là tối ưu hóa việc tìm kiếm thông tin quan trọng và giảm thiểu việc bỏ qua thông tin không cần thiết. Encoder-Decoder là một nhân vật quan trọng, anh ta sở hữu khả năng biến đổi một câu văn bản dài thành một biểu tượng được giảm chiều, sau đó dựa trên biểu tượng đó để tạo ra một câu văn bản mới. Điểm ưu của anh ta là tính chất học hỏi của mình và khả năng biến đổi câu văn bản một cách tự nhiên. Nhược điểm là sự phức tạp của mô hình và sự tốn kém trong việc học hỏi.

Các trường phái luôn phải tìm cách sử dụng ưu điểm của mỗi anh hùng để khắc phục nhược điểm của nhau để chống lại lũ quái vật. Nền hoà bình của thế giới machine learnng vẫn được duy trì bởi sự hợp tác và học hỏi liên tục của tất cả nhưng nhân vật trong thế giới.

Vì sao chọn hàm MSE (Mean Squared Error) cho mô hình Linear Regression.

Problem	Why	Solution
Đánh giá độ sai lệch giữa giá trị dự đoán và giá trị thực tế.	Dùng để cập nhật trọng số để mô hình dự đoán giá trị tốt nhất.	$\mathcal{L}(\theta; x, y) = \sum_{i=1}^{n} (\tilde{y}^{(i)} - y^{(i)})$
Trả về giá trị âm khi giá trị dự đoán nhỏ hơn giá trị thực tế.	Hàm độ lỗi thể hiện sự sai khác giữa giá trị dự đoán và giá trị thực tế, nên về mặt ý nghĩa độ lỗi phải là một hàm không âm.	$\mathcal{L}(\theta; x, y) = \sum_{i=1}^{n} \lvert\tilde{y}^{(i)} - y^{(i)}\rvert$
Không thể đạo hàm tại 0.	Hàm độ lỗi phải có thể tính được đạo hàm.	$\mathcal{L}(\theta; x, y) = \sum_{i=1}^{n} (\tilde{y}^{(i)} - y^{(i)})^2$
Phụ thuộc vào số lượng mẫu dữ liệu.	Nếu không chia trung bình cho số mẫu, ta sẽ khó xác định được độ lỗi hiện tại của mô hình có áp dụng được trong thực tết hay không.	$\mathcal{L}(\theta; x, y) = \frac{1}{n}\sum_{i=1}^{n} (\tilde{y}^{(i)} - y^{(i)})^2$
	Chia thêm 2 để sau này bước tính đạo hàm có công thức đẹp hơn.	$\mathcal{L}(\theta; x, y) = \frac{1}{2n}\sum_{i=1}^{n} (\tilde{y}^{(i)} - y^{(i)})^2$

Vì sao sử dụng hàm Binary Cross Entropy mà không sử dụng MSE cho hàm loss của Logistic Regression.

Trong Logistic Regression, ta dự đoán xác suất của một đối tượng thuộc vào một lớp nhất định. Do đó, kết quả dự đoán của chúng ta sẽ là một giá trị xác suất giữa 0 và 1. Nếu dùng MSE thì sai số là quá thấp.

BCE: $\mathcal{L}(\theta; x, y) = -(y log\tilde{y} + (1 - y)\log(1 - \tilde{y}))$

Đảm bảo các tính chất của hàm lỗi (có thể đạo hàm, giá trị nhỏ hơn khi dự đoán gần với giá trị thức tế và có giá trị lớn hơn khi dự đoán xa với giá trị thực tế $+\infty$.

Thành phần của CNN:

- Hàm tích chập (convolution)
- Hàm phi tuyến (non-linear gating)
- Hàm chiết xuất (pooling)
- Hàm chuẩn hoá (normalization)
- Hàm kết nối đầy đủ (fully connected)
- Hàm softmax.

4 cái đầu học cách biểu dễn đặc trưng của dữ liệu, 2 cái sau phân loại đặc trưng.

Một số công thức cần lưu ý.

MSE: $\mathcal{L}(\theta; x, y) = \frac{1}{2n}\sum_{i=1}^{n}\left(\tilde{y}^{(i)} - y^{(i)}\right)^2$

BCE: $\mathcal{L}(\theta; x, y) = -(ylog\tilde{y} + (1-y)\log(1-\tilde{y}))$

Sigmoid: $\sigma(z) = \frac{1}{1+e^{-z}}$

Softmax: $\tilde{y} = \frac{e^{z}i}{\sum_{k=1}^{K} e^{z}k}$

Số lượng tham số của NN: $(I+1) * N$

Số lượng tham số của CNN: $(K * K * D + 1) * Số\ Kernel$

Output sau Conv layer: $W = \frac{W-K+2P}{S} + 1$

Output sau Pooling layer: $W = \frac{I-F}{S} + 1$

Một số vấn đề cần giải quyết của các mô hình học sâu:

- Triệt tiêu đạo hàm
- Số lượng tham số tăng
- Số lượng phép tính tăng
- Đảm bảo tính tổng quát của mô hình

Overfitting là gì? Nguyên nhân? Cách khắc phục?

Overfitting là hiện tượng mô hình học thuộc lòng dữ liệu huấn luyện làm mất đi tính tổng quát.

Nguyên nhân:

- Mô hình phức tạp: mô hình với nhiều tham số yêu cầu nguồn dữ liệu lớn.
- Dữ liệu không đủ: nếu không đủ dữ liệu huấn luyện để biểu thị các mẫu cơ bản trong dữ liệu, mô hình sẽ khớp quá mức với dữ liệu hạn chế mà nó đã thấy.
- Điểm nhiễu: khiến mô hình khớp với nhiễu ngẫu nhiên trong dữ liệu thay vì các mẫu cơ bản.

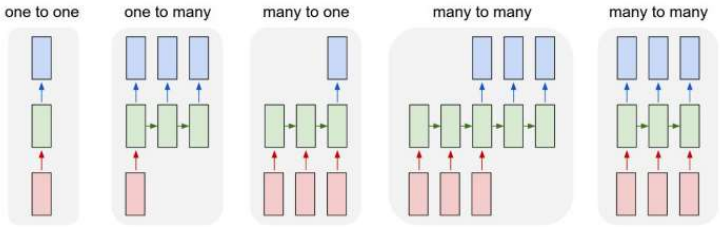
Cách khắc phục:

- Regularization: Thêm thời hạn phạt cho hàm mất mát ngăn mô hình khớp với nhiễu.
- Cross-Validation: Kỹ thuật huấn luyện mô hình trên tập con của dữ liệu và đánh giá trên tập còn lại.
- Early Stopping: Dừng việc huấn luyện khi độ chính xác trên tập valid bắt đầu giảm.
- Simplifing: Đơn giản hoá mô hình.

Mô hình	Đặc trưng
AlexNet	Sử dụng hàm phi tuyến ReLU thay cho sigmoid hoặc tanh giúp tốc độ nhanh hơn. Sử dụng tăng cường dữ liệu. Sử dụng kỹ thuật Dropout giúp tăng tính tổng quát của mô hình, giảm thời gian huấn luyện.
VGG-16	Thay gì thay phiên conv+pooling thì sử dụng liên tục conv giúp giảm số tham số.
ResNet	Sử dụng Batch Normalization giải quyết hiện tượng triệt tiêu/ bùng nổ gradient, chưa giải quyết hiện tượng bão hoà loss. (không của resnet) Sử dụng khối nối tắt giúp giữ được thông tin quan trọng và giảm số bước làn truyền ngược.
DenseNet	Mạng CNN truyền thống thì L lớp liên tiếp có L kết nối, còn DenseNet sẽ có $\frac{L(L+1)}{2}$ kết nối. Tác dụng như ResNet.
MobileNet	Cấu thành từ nhều lớp depthwise separable convolution, từ input ra feature map trải qua 2 bước (depthwise, pointwise) giúp giảm số tham số, giúp mô hình nhẹ và tính toán nhanh hơn mà vẫn giữ độ chính xác cao. Chi phí tính toán thông thường $D_K D_K M D_F D_F N$ Đối với MobileNet $D_K D_K M D_F D_F + M N D_F D_F$

Ảnh: thông tin có mối quan hệ về không gian.

Văn bản: thông tin có mối quan hệ về trình tự.

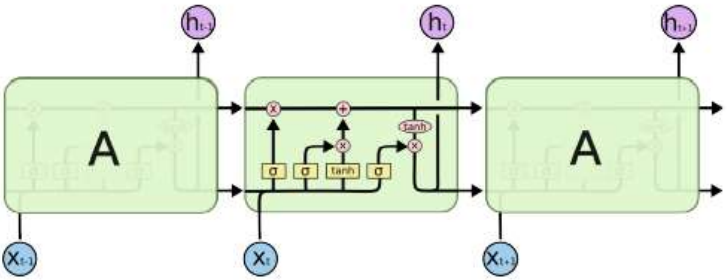


Hình 5.3: Một số dạng biến thể của mạng RNN.

Biến thể của mạng RNN:

- One-to-one: phân loại hình ảnh.
- One-to-many: mô tả dữ liệu ảnh.
- Many-to-one: phân loại văn bản
- Many-to-many: 2 dạng
    - Dịch tự động văn bản
    - Gán nhãn từ loại

Vấn đề phụ thuộc xa: thông tin bị tiêu biến dần sau nhiều lần nạp dữ liệu vào các trạng thái của mạng.



Hình 5.5: Kiến trúc một ô nhớ của mạng LSTM.

LSTM: trục thông tin ngữ cảnh.