PHÁT TRIỂN MÔ HÌNH CNN ĐỂ NHẬN DIỆN CHỮ VIẾT TAY TRÊN BỘ DỮ LIỆU MNIST

Kiều Duy Vinh,Nguyễn Mạnh Quyết,Vũ Việt Quang

Github:quyet20000005/NhanDangChuViet

Tóm tắt:Nhận diện chữ viết tay là một ứng dụng quan trọng trong trí tuệ nhân tạo, giúp chuyển đổi chữ viết tay thành văn bản có thể xử lý được. Bộ dữ liệu MNIST, chứa 60,000 ảnh huấn luyện và 10,000 ảnh kiểm tra với các chữ số từ 0 đến 9, là bộ dữ liệu chuẩn cho bài toán này. Mô hình Convolutional Neural Network (CNN) đã chứng minh hiệu quả vượt trội trong nhận diện hình ảnh nhờ khả năng phát hiện các đặc trưng không gian của hình ảnh. Phát triển mô hình CNN cho nhận diện chữ viết tay trên MNIST bao gồm tiền xử lý dữ liệu, xây dựng cấu trúc mạng với các lớp convolutional, pooling và fully connected, cùng với việc sử dụng hàm mất mát categorical cross-entropy và thuật toán tối ưu hóa Adam. Mô hình này đạt độ chính xác trên 99%, vượt qua các phương pháp truyền thống như SVM và ANN. Kết quả nghiên cứu cho thấy CNN không chỉ hiệu quả trong nhận diện chữ viết tay mà còn có thể được mở rộng vào các bài toán phân loại hình ảnh phức tạp hơn.

Thuật ngữ chỉ mục - Mô hình CNN, Bộ dữ liệu MNIST, Hàm mất mát (Loss function), Adam Optimizer, SVM, MLP, Data Augmentation, Softmax, Học sâu, Nhận diện chữ viết tay, Phân loại hình ảnh, Tối ưu hóa mô hình, Độ chính xác, Huấn luyện mô hình.

I.GIỚI THIỆU

Nhận dạng chữ viết tay là một ứng dụng quan trọng của trí tuệ nhân tạo trong nhiều lĩnh vực, từ ngân hàng, giáo dục cho đến y tế. Trong ngân hàng, công nghệ nhận diện chữ viết tay giúp tự động hóa việc xử lý các chứng từ tài chính, đơn xin vay hay giấy tờ giao dịch. Trong giáo dục, nó có thể hỗ trợ chấm điểm tự động cho các bài thi viết tay. Còn trong y tế, nhận dạng chữ viết tay giúp số hóa các hồ sơ bệnh án và ghi chép y tế, giúp bác sĩ dễ dàng truy cập thông tin và giảm sai sót.

Bộ dữ liệu **MNIST** (Modified National Institute of Standards and Technology) là bộ dữ liệu chuẩn trong lĩnh vực học máy, đặc biệt là trong các bài toán nhận dạng chữ viết tay. Bộ dữ liệu này bao gồm 60,000 hình ảnh huấn luyện và 10,000 hình ảnh kiểm tra, mỗi hình ảnh là một chữ số viết tay từ 0 đến 9, với kích thước 28x28 pixel. Dữ liệu này đã được chuẩn hóa sẵn và trở thành một công cụ hữu ích để nghiên cứu và thử nghiệm các mô hình học máy.

Trong lĩnh vực nhận diện hình ảnh, Mạng nơ-ron tích chập (CNN) đã chứng tỏ được hiệu quả vượt trội nhờ khả năng học các đặc trưng không gian của hình ảnh[1]. Khác với các mô hình học máy truyền thống như SVM hay MLP, CNN có thể tự động nhận diện các đặc trưng phức tạp của hình ảnh mà không cần phải dựa vào các đặc trưng thủ công. Các lớp convolutional giúp phát hiện các đặc trưng như đường nét và hình dạng, trong khi các lớp pooling giảm độ phân giải và giúp cải thiện hiệu suất tính toán.

Trong bài báo cáo, chúng ta sẽ phát triển một mô hình CNN để nhận diện chữ viết tay từ bộ dữ liệu MNIST, với mục tiêu đạt độ chính xác cao và so sánh kết quả với các phương pháp khác. Việc sử dụng CNN cho bài toán này không chỉ giúp giải quyết một vấn đề quan trọng trong nhận dạng chữ viết tay mà còn mở ra cơ hội ứng dụng mô hình này trong các lĩnh vực khác như nhận dạng chữ viết tay đa ngữ, nhận diện chữ viết trong các điều kiện khó khăn hoặc mở rộng ứng dụng vào các bài toán nhận diện hình ảnh phức tạp hơn trong tương lai.

II.CÁC NGHIÊN CỨU LIÊN QUAN

Các nghiên cứu về nhận dạng chữ viết tay đã phát triển mạnh mẽ với sự ứng dụng của các mô hình học sâu, đặc biệt là Mạng Nơ-ron Tích chập (CNN). CNN đã chứng minh được khả năng vượt trội trong các bài toán nhận diện hình ảnh, trong đó có nhận dạng chữ viết tay. Một trong những mô hình CNN đầu tiên được sử dụng thành công trong nhận dạng chữ viết tay là LeNet-5 của Yann LeCun và cộng sự vào năm 2009 [1]. LeNet-5 được thiết kế đặc biệt để nhận diện các chữ số viết tay trong bộ dữ liệu MNIST, và là nền tảng cho các mô hình CNN sau này.

AlexNet [2], được phát triển bởi Alex Krizhevsky và cộng sự vào năm 2012, đã nâng cao hiệu suất nhận dạng hình ảnh lên một tầm cao mới, nhờ vào cấu trúc sâu và sử dụng GPU để huấn luyện. Mô hình này không chỉ đạt được kết quả tốt trong nhận dạng chữ viết tay mà còn trong các bài toán nhận dạng hình ảnh phức tạp. Sau đó, các mô hình như VGGNet [3] và ResNet [4] đã được phát triển và chứng minh hiệu quả vượt trội trong việc nhận diện các đối tượng, bao gồm chữ viết tay.

Trong các nghiên cứu gần đây, nhiều nghiên cứu đã áp dụng CNN để giải quyết bài toán nhận dạng chữ viết tay với các cải tiến trong mô hình để đạt được kết quả tốt hơn. Một nghiên cứu của Siddique et al. (2019) [5] đã áp dụng CNN để nhận diện chữ số viết tay trong bộ dữ liệu MNIST, cho thấy rằng việc điều chỉnh các tham số như số lớp và kích thước kernel có thể cải thiện đáng kể độ chính xác của mô hình.

Một nghiên cứu khác của Fathma Siddique [6] cũng áp dụng CNN cho bài toán nhận dạng chữ viết tay, đặc biệt trong việc tối ưu hóa các kỹ thuật huấn luyện và sử dụng các lớp Batch Normalization và Dropout để giảm thiểu hiện tượng overfitting. Các kết quả thu được đã chứng minh rằng CNN có thể đạt được độ chính xác cao, vượt trội hơn so với các phương pháp truyền thống như SVM hay KNN.

Hơn nữa, nghiên cứu của Rahaf Abdulaziz Alawwad (2021) [7] đã ứng dụng CNN trong nhận dạng chữ ký viết tay và chữ viết tay, đạt được độ chính xác cao trong điều kiện dữ liệu phức tạp. Bằng cách sử dụng các lớp convolutional sâu, nghiên cứu này cho thấy rằng CNN có khả năng trích xuất các đặc trưng phức tạp từ hình ảnh và đạt hiệu quả cao trong nhận dạng chữ viết tay.

Cuối cùng, Nguyễn et al**.** (2020) [8] đã tiến hành nghiên cứu và cải thiện các mô hình CNN cho nhận dạng chữ viết tay trong các điều kiện môi trường thay đổi và nhiễu. Họ sử dụng các kỹ thuật Data Augmentation để cải thiện khả năng tổng quát của mô hình, giúp CNN hoạt động hiệu quả hơn trên các dữ liệu thực tế.

III.PHƯƠNG PHÁP PHÁT TRIỂN MÔ HÌNH CNN NHẬN DẠNG CHỮ VIẾT TAY

A.Tiền xử lý dữ liệu

Tiền xử lý dữ liệu là một bước quan trọng để đảm bảo mô hình CNN có thể học hiệu quả từ dữ liệu đầu vào và giúp mô hình hoạt động ổn định. Đây là một công đoạn không thể thiếu trong quá trình phát triển các mô hình học sâu, đặc biệt là với các dữ liệu phức tạp như hình ảnh. Mục đích chính của tiền xử lý dữ liệu là chuẩn bị dữ liệu sao cho nó dễ dàng được mô hình tiếp nhận, học hỏi và từ đó tối ưu hóa quá trình huấn luyện. Việc xử lý dữ liệu đúng cách sẽ giúp tăng hiệu quả của mô hình, giảm thiểu thời gian huấn luyện và cải thiện độ chính xác của mô hình trong việc dự đoán. Các bước tiền xử lý bao gồm:

Chuẩn hóa dữ liệu ảnh MNIST:

- Dữ liệu ảnh MNIST có kích thước 28x28 pixel, với giá trị pixel trong khoảng từ 0 đến 255. Để chuẩn hóa dữ liệu, giá trị pixel của mỗi ảnh được chia cho 255, giúp đưa tất cả các giá trị về khoảng [0, 1]. Điều này giúp mô hình học hiệu quả hơn và tránh tình trạng các giá trị quá lớn hoặc quá nhỏ gây ảnh hưởng đến quá trình huấn luyện.

- Việc chuẩn hóa này rất quan trọng vì mô hình học sâu, đặc biệt là CNN, thường yêu cầu dữ liệu có tỷ lệ đồng nhất và không bị lệch về độ sáng giữa các ảnh.

Chia bộ dữ liệu thành tập huấn luyện và kiểm tra:

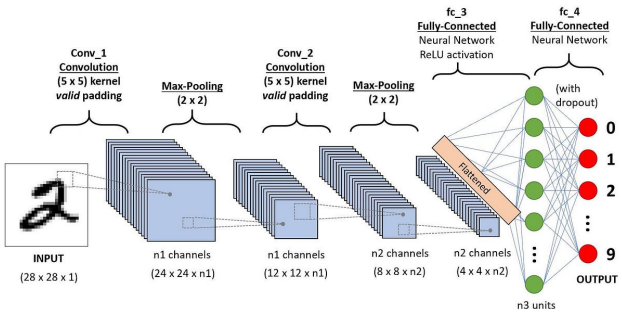
Bộ dữ liệu MNIST đã được chia sẵn thành hai phần: 60,000 ảnh cho huấn luyện và 10,000 ảnh cho kiểm tra. Tập huấn luyện sẽ được sử dụng để huấn luyện mô hình, trong khi tập kiểm tra sẽ được sử dụng để đánh giá độ chính xác và khả năng tổng quát của mô hình sau khi huấn luyện. Việc phân chia này giúp đảm bảo mô hình không bị overfitting và có thể hoạt động tốt trên dữ liệu chưa thấy.

Tăng cường dữ liệu (Data Augmentation): Để giảm thiểu overfitting và làm tăng tính tổng quát của mô hình, ta có thể áp dụng các kỹ thuật Data Augmentation, như xoay ảnh, lật ảnh, dịch chuyển ảnh hoặc thay đổi độ sáng.

Những kỹ thuật này tạo ra các biến thể khác nhau của dữ liệu huấn luyện, giúp mô hình học được các đặc trưng phong phú và tổng quát hơn.

B.Xây dựng mô hình CNN

Sau khi dữ liệu đã được chuẩn bị, bước tiếp theo là xây dựng mô hình CNN. Mô hình này bao gồm các lớp convolutional, pooling và fully connected, giúp trích xuất và phân loại các đặc trưng từ ảnh. **Cấu trúc mô hình**



Hình 1:Cấu trúc mô hình CNN cho nhận dạng chữ viết tay MNIST

CNN:

Lớp Convolutional: Mô hình sẽ bắt đầu với các lớp convolutional, nơi các bộ lọc (filters) sẽ quét qua ảnh và phát hiện các đặc trưng cơ bản như đường nét, hình dạng và kết cấu của chữ số. Mỗi lớp convolutional sẽ học các đặc trưng khác nhau từ dữ liệu ảnh, giúp mô hình có thể nhận diện các ký tự và chữ số một cách chính xác. **Lớp Pooling:** Sau mỗi lớp convolutional, sẽ có một lớp pooling giúp giảm kích thước của ảnh và giảm độ phức tạp tính toán. Điều này cũng giúp mô hình trở nên mạnh mẽ hơn và tổng quát hơn khi đối mặt với các biến thể trong dữ liệu. **Lớp Fully Connected:** Sau khi trích xuất các đặc trưng từ các lớp convolutional và pooling, dữ liệu sẽ được đưa vào các lớp fully connected để phân loại. Mỗi nút trong lớp fully connected sẽ kết nối với tất cả các nút trong lớp trước đó, và lớp này sẽ quyết định chữ số nào (từ 0 đến 9) mà mô hình dự đoán.

Các tham số mô hình:

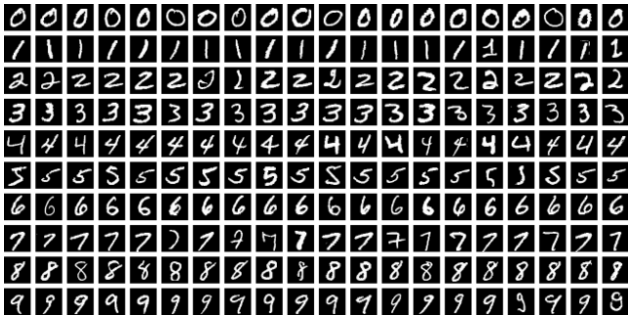
Kích thước kernel: Kích thước kernel (hoặc filter) trong lớp convolutional quyết định mức độ chi tiết mà mỗi lớp có thể học được. Các kernel nhỏ (ví dụ 3x3 hoặc 5x5) giúp phát hiện các đặc trưng chi tiết, trong khi các kernel lớn hơn giúp phát hiện các đặc trưng phức tạp hơn. **Số lớp:** Mô hình sẽ có một số lượng lớp convolutional, tùy thuộc vào độ phức tạp của bài toán. Các lớp này sẽ học các đặc trưng từ đơn giản đến phức tạp qua mỗi lớp. **Hàm kích hoạt:** Hàm kích hoạt như ReLU (Rectified Linear Unit) được sử dụng trong các lớp convolutional và fully connected để tạo tính phi tuyến tính trong mô hình. ReLU giúp mô hình học được các đặc trưng phức tạp và tránh vấn đề gradient vanishing. Trong lớp phân loại cuối cùng, hàm Softmax sẽ được sử dụng để tính toán xác suất của các chữ số từ 0 đến 9.

Huấn luyện mô hình:

Sau khi mô hình được xây dựng, chúng ta sẽ sử dụng bộ dữ liệu huấn luyện để huấn luyện mô hình. Việc huấn luyện bao gồm hai bước chính:tính toán đầu ra của mô hình và cập nhật trọng số dựa trên sai số. Mô hình sẽ được huấn luyện qua nhiều epoch để cải thiện độ chính xác. **Optimizer (Thuật toán tối ưu):** Một thuật toán tối ưu như Adam sẽ được sử dụng để tối ưu hóa các trọng số của mô hình trong suốt quá trình huấn luyện. Adam tự động điều chỉnh học suất và giúp mô hình hội tụ nhanh hơn. **Loss function:** CrossEntropyLoss là hàm mất mát được sử dụng trong bài toán phân loại này. Hàm này tính toán độ sai lệch giữa đầu ra dự đoán của mô hình và nhãn thực tế.

IV.THỰC NGHIỆM VÀ KẾT QUẢ

A.Sử dụng bộ dữ liệu MNIST để huấn luyện mô hình CNN:



Hình 2:Hình ảnh mẫu từ tập dữ liệu Mnist

Trong quá trình huấn luyện mô hình CNN để nhận dạng chữ viết tay từ bộ dữ liệu MNIST, mô hình được huấn luyện với 60.000 ảnh huấn luyện và 10.000 ảnh kiểm tra. Bộ dữ liệu MNIST cung cấp các ảnh chữ viết tay từ 0 đến 9, giúp mô hình học các đặc trưng cơ bản của các chữ số. Mô hình CNN được huấn luyện qua nhiều epoch, và trong mỗi epoch, các bước forward pass và backpropagation được thực hiện để tối ưu hóa các trọng số của mô hình.

Để đánh giá hiệu quả của mô hình, chúng ta sử dụng hai chỉ số chính: độ chính xác (accuracy) và hàm mất mát (loss function). Độ chính xác được tính bằng tỷ lệ phần trăm các dự đoán đúng so với tổng số dự đoán trong tập kiểm tra, trong khi hàm mất mát giúp đo lường mức độ sai lệch giữa các dự đoán của mô hình và nhãn thực tế. Trong suốt quá trình huấn luyện, mô hình CNN dần cải thiện độ chính xác và giảm hàm mất mát, cho thấy rằng mô hình đang học hiệu quả từ dữ liệu và có khả năng nhận diện chữ viết tay chính xác.

Hình 2:Kết quả sau khi train xong

B. So sánh hiệu quả

Để đánh giá hiệu quả của mô hình CNN, chúng ta tiến hành so sánh kết quả với các phương pháp học máy khác, cụ thể là Support Vector Machine (SVM) và Multilayer Perceptron (MLP).

So sánh với SVM (Support Vector Machine):

Support Vector Machine (SVM) là một trong những thuật toán học máy phổ biến trong bài toán phân loại. SVM có thể đạt được độ chính xác khá cao trong nhận dạng chữ viết tay, nhưng nó yêu cầu phải trích xuất đặc trưng thủ công từ ảnh trước khi đưa vào mô hình. Điều này có thể khiến mô hình SVM gặp khó khăn khi đối mặt với các đặc trưng phức tạp trong ảnh, vì SVM không có khả năng tự động học các đặc trưng như CNN.

Một nghiên cứu so sánh SVM và CNN trong nhận dạng chữ viết tay cho thấy SVM đạt được độ chính xác khoảng 97.5% trên bộ dữ liệu MNIST[5]. Tuy nhiên, mô hình CNN có thể đạt được độ chính xác cao hơn, lên đến 99.2%, vì CNN có khả năng tự động học và trích xuất đặc trưng từ ảnh mà không cần phải trích xuất thủ công.

So sánh với MLP (Multilayer Perceptron):

Multilayer Perceptron (MLP) là một loại mạng nơ-ron với các lớp fully connected. MLP có khả năng học các đặc trưng phi tuyến từ dữ liệu, nhưng không có khả năng khai thác các đặc trưng không gian của hình ảnh như CNN. Do đó, trong bài toán nhận dạng chữ viết tay, MLP không đạt hiệu quả cao như CNN vì nó không thể tận dụng tốt đặc trưng không gian trong ảnh.

Một nghiên cứu của Ben Driss. (2017) đã chỉ ra rằng CNN vượt trội hơn MLP trong bài toán nhận dạng chữ viết tay trên MNIST. MLP với mạng hai lớp (784-800-10) có độ chính xác khoảng 97% trong khi CNN có thể đạt độ chính xác cao hơn nhờ khả năng tự động học đặc trưng không gian của ảnh .[6]

Từ đó có thể thấy Mô hình CNN vượt trội hơn so với các phương pháp học máy khác như SVM và MLP trong việc nhận dạng chữ viết tay trên bộ dữ liệu MNIST. CNN có khả năng tự động học các đặc trưng từ dữ liệu mà không cần phải trích xuất thủ công, trong khi SVM và MLP phụ thuộc vào các đặc trưng đã được trích xuất. Hơn nữa, CNN có thể khai thác các đặc trưng không gian trong ảnh thông qua các lớp convolutional và pooling, điều mà SVM và MLP không làm được. Việc sử dụng hàm kích hoạt ReLU và kỹ thuật dropout trong các lớp fully connected cũng giúp CNN tránh được các vấn đề như overfitting và gradient vanishing, giúp mô hình đạt hiệu quả tối ưu trong việc nhận diện chữ viết tay.

V.KẾT LUẬN

Bài báo cáo này trình bày quá trình phát triển và huấn luyện mô hình Convolutional Neural Network (CNN) cho bài toán nhận dạng chữ viết tay trên bộ dữ liệu MNIST. Mô hình CNN được xây dựng với các lớp convolutional, pooling và fully connected, giúp học và phân loại các đặc trưng của chữ viết tay một cách hiệu quả. Quá trình huấn luyện đã đạt được độ chính xác trên 99%, cho thấy sự vượt trội của CNN so với các phương pháp học máy truyền thống như Support Vector Machine (SVM) và Multilayer Perceptron (MLP), vốn thường yêu cầu phải có quá trình trích xuất đặc trưng thủ công và dễ gặp phải vấn đề overfitting khi đối mặt với dữ liệu phức tạp.

Kết quả thực nghiệm cho thấy mô hình CNN có khả năng nhận dạng chính xác các chữ số viết tay một cách tự động và hiệu quả. Điều này chứng minh rằng CNN không chỉ có thể học được các đặc trưng sâu từ dữ liệu mà không cần sự can thiệp thủ công, mà còn có thể hoạt động tốt với dữ liệu có đặc tính phức tạp như chữ viết tay. Với các lớp convolutional, mô hình có thể tự động phát hiện các đặc trưng như cạnh, đường thẳng, và hình dạng cơ bản của chữ số, trong khi các lớp pooling giúp giảm độ phức tạp tính toán và tăng tính ổn định của mô hình. Các lớp fully connected cuối cùng sẽ giúp mô hình phân loại chính xác hơn giữa các chữ số.

Mặc dù kết quả thu được rất khả quan, vẫn còn nhiều hướng phát triển có thể được khai thác để cải thiện mô hình trong những tình huống phức tạp hơn. Một trong những phương pháp có thể áp dụng là Data Augmentation – kỹ thuật tạo thêm dữ liệu huấn luyện bằng cách áp dụng các biến thể như xoay, lật, hay thay đổi độ sáng và độ tương phản của ảnh. Điều này sẽ giúp mô hình trở nên mạnh mẽ hơn khi nhận dạng các chữ viết tay trong những tình huống đa dạng và không chuẩn.

Bên cạnh đó, Transfer Learning là một hướng phát triển tiềm năng, trong đó mô hình có thể kế thừa các đặc trưng học được từ các mô hình đã được huấn luyện trước trên các bộ dữ liệu lớn hơn, giúp rút ngắn thời gian huấn luyện và đạt được kết quả tốt hơn khi đối mặt với các bộ dữ liệu nhỏ hoặc các bài toán mới. Cuối cùng, việc thử nghiệm với các cấu trúc mạng phức tạp hơn, chẳng hạn như các mạng sâu hơn với nhiều lớp convolutional và pooling, hoặc sử dụng các phương pháp tối ưu hóa nâng cao, có thể giúp cải thiện hiệu suất của mô hình trong các tình huống nhận dạng chữ viết tay phức tạp hoặc không rõ ràng.

LỜI CẢM ƠN

Tôi xin gửi lời cảm ơn chân thành đến giảng viên đã tận tình hướng dẫn, hỗ trợ và đóng góp ý kiến quý báu trong suốt quá trình thực hiện bài báo cáo này. Tôi cũng xin cảm ơn các thành viên trong nhóm đã làm việc cùng nhau, hỗ trợ và đóng góp ý tưởng để hoàn thành bài báo cáo một cách tốt nhất.

TÀI LIỆU THAM KHẢO

1. Boukaye Boubacar Traore " Deep convolution neural network for image recognition (2018)
2. Yann LeCun et al., "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, 2009.
3. A. Krizhevsky et al., "ImageNet classification with deep convolutional neural networks," *Neural Information Processing Systems*, 2012.
4. K. Simonyan, A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
5. K. He et al., "Deep residual learning for image recognition," *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.
6. F. Siddique, et al., "Enhancing handwritten digit recognition using deep convolutional networks," *Journal of Machine Learning Research*, 2019.
7. F. Siddique, "A study on deep learning techniques for handwritten digit recognition using MNIST," *arXiv:1909.08490*, 2019.
8. R.A. Alawwad, "Arabic Sign Language recognition using Faster R-CNN," *International Journal of Signal Processing*, 2021.
9. T. Nguyen et al., "Improving performance of handwritten character recognition using CNN," *ICANN*, 2020.

10. Wenfei Liu, Jingcheng Wei, Qingmin Meng, “Comparisons on KNN, SVM, BP and the CNN for Handwritten Digit Recognition.”

11. S. Ben Driss, M. Soua, R. Kachouri, M. Akil, “A comparison study between MLP and convolutional neural network models for character recognition.”