Smart Closet: Interactive Machine Learning for Personalized Wardrobe Management

JIHYUN PARK

University Paris-Saclay, Human Computer Interaction, M1

YING LAI

University Paris-Saclay, Artificial Intelligence, M1

WENTING BAI

University Paris-Saclay, Human Computer Interaction, M1

1. Scenario

This project endeavors to address the needs of individuals seeking a comprehensive digital representation of their wardrobe, offering them a tool to not only organize their clothing items but also gain deeper insights into their personal style preferences. By leveraging the power of machine learning, we aim to provide users with an automated solution for categorizing their garments, thereby streamlining the process of managing their wardrobe inventory.

One of the key features of this system is its ability to automatically classify clothing items based on their visual characteristics, thanks to the implementation of a sophisticated machine learning model. Through the intuitive interface of the "Add Clothes" tab, users can effortlessly upload images of their garments by simply dragging and dropping them onto the platform. Subsequently, the machine learning algorithm analyzes these images and assigns appropriate labels to each item, effectively organizing the wardrobe with minimal user input.

However, we understand that individual style preferences can vary greatly, and a one-size-fits-all approach may not always suffice. To address this, we have incorporated customizable labeling options, allowing users to tailor the classification system to better align with their unique style sensibilities. Whether it's assigning specific labels to reflect preferred color schemes, fabric types, or garment styles, users have the flexibility to create personalized categories that resonate with their individual fashion preferences.

Moreover, we recognize the importance of user feedback in refining and enhancing the performance of our machine learning model. Hence, we have implemented a robust feedback mechanism that enables users to provide real-time input on the accuracy of the system's predictions. Should the assigned labels not accurately reflect the user's perception of a particular item, they can easily modify them using the dropdown menu or opt to create custom labels from scratch. These user-initiated adjustments are then documented in the "Error" tab, allowing administrators to review and incorporate them into the training dataset for ongoing model refinement.

For administrators overseeing the system, additional functionalities are available to facilitate the management and optimization of the machine learning pipeline. The "Training Data" tab provides administrators with the capability to upload and update the training dataset, ensuring that the model remains current and responsive to evolving fashion trends. Moreover, administrators have the option to define custom labels for specific instances, further enhancing the system's adaptability and accuracy.

In essence, this project represents a collaborative effort between users and administrators, with the shared goal of creating a robust and user-centric solution for wardrobe management. By harnessing the power of machine learning and user feedback, we strive to empower individuals to take control of their personal style journey, one garment at a time.

1. Literature review

2.1.Convolutional neural network

The work from LeCun named “Gradient-Based learning Applied to Document Recognition”[[1]](#bookmark=kix.ao8v6pl3cqn2) is the first to successfully use convolutional neural network in the field of image recognition and classification. The Convolutional Neural Network (CNN) stands out as a widely employed and notably effective artificial neural network in the realm of image recognition and classification. Its architecture typically comprises three primary layers: the convolutional layer, pooling layer, and fully-connected layer, all combined to shape the CNN's structure.[[2]](#bookmark=kix.gjcq0g9lt7zp) The convolutional layer serves as the foundational element within a convolutional neural network (CNN), undertaking the bulk of computational tasks.[[3]](#bookmark=kix.hk9ppb859jc0) Within this layer, arrays of weights, referred to as filters, traverse the input image, performing element-wise multiplication with each pixel value. This operation, applied across the entire image, serves as a feature extractor, identifying patterns and structures within the input. Following this, the pooling layer progressively diminishes the spatial dimensions of the representation, curbing parameter count and computational load to prevent overfitting. Lastly, the fully connected layer amalgamates the output from convolutional and pooling layers, encapsulating the high-level features extracted from the input image.

2.2.Fashion and image classification

Liu et al. introduced DeepFashion, a vast clothing dataset encompassing over 800,000 images featuring diverse scenarios, clothing brands, and a plethora of attributes.[[4]](#bookmark=kix.xz20tf4a5q6n) Notably, the dataset boasts meticulous annotation, with professional annotators meticulously assigning images to their respective categories. This richly annotated resource serves as a catalyst for numerous research endeavors. For instance, in the paper "Apparel Classification with Style" by Bossard et al. a comprehensive pipeline is presented, targeting the recognition and classification of clothing worn by individuals in "natural scenes."[[5]](#bookmark=kix.sh46ycj2as05) In another investigation titled "Machine Fashion: Artificial Intelligence Based Clothing Fashion Stylist" by Haosha Wang, the researcher delved into the question of whether an AI system could serve as a fashion stylist.[[6]](#bookmark=kix.26on6gfgget4) Wang conducted a survey study and developed an application called Style-Me, which utilizes a Multilayer Perceptron model to learn user preferences and provide fashion recommendations or "styles."

1. Design

This project endeavors to address the needs of individuals seeking a comprehensive digital representation of their wardrobe, offering them a tool to not only organize their clothing items but also gain deeper insights into their personal style preferences. The design of the proposed clothing classifier system is inspired by a comprehensive literature review and insights gathered from meticulous data collection and auditing processes. The literature emphasizes the importance of interactive machine learning systems that not only learn from their initial training data but also continuously evolve through user interactions. This design philosophy underpins our system, enabling it to adapt and improve over time as it receives feedback from users.

3.1.Interface design

The interface is designed to be user-friendly and intuitive, facilitating easy navigation through various functionalities. We designed a total of 8 pages. The first four tabs: 'Training set', 'Training', 'Testing' and 'Error reports' are designed for administrators, and the last four tabs are 'My closet', 'Add clothes', 'Visualization' tabs are used by users. If administrators and users have no idea how to use this app, they can get useful information in the ‘Guide’ page. ‘Training set' is where all images are stored and includes the labels of the images. Administrators can train the classification model using datasets in ‘training set’ and evaluate the model by dragging other clothes pictures to the ‘Testing’ tab, then they can decide if they want to optimize the model, like modifying the parameters or trying other classification methods. In the page 'Error reports', administrators decide whether to upload an image provided by the users to the complete datasets after receiving feedback from the users. 'My closet', 'Add clothes', 'Visualization' and 'Guide' tabs used by users.

3.2. Functions

Here are some of the functions we designed for this project:

Data Collections: The "Training Data" tab is straightforward, encouraging users to contribute to the dataset actively. The process of updating training data is seamless, with users able to upload images and assign labels effortlessly. This interaction is crucial for amassing a rich and varied dataset, which is fundamental for training a robust classifier.

Model training and testing (for administrators): The "Training" tab provides users with the capability to influence the training process by selecting classifier parameters such as layers, epochs, and batchSize. This feature is designed to engage users with varying levels of expertise, offering them a sense of control over the model's training regime. Testing is made accessible through the "Testing" tab, where administrators upload new images for classification, mirroring the data collection process. This consistency in design between training and testing phases simplifies user interaction, making the system more approachable. The inclusion of a confusion matrix in the testing results provides detailed insights into the model's performance, highlighting areas where improvements are necessary.

Model training and testing (for users): In the ‘Add clothes’ tab, users upload photos, and the system automatically classifies them based on the trained model.Users can modify the categories of classified photos and feedback this modification operation to the administrator to improve the algorithm.

Error analysis and user participation: The "Error reports" tab is a novel feature that distinguishes our system. It allows users to audit the classifier's predictions and make corrections, which are then reflected in the "Error set" box. This mechanism for feedback and correction is pivotal for iterative improvement of the classifier. By examining confidence bar graphs for each image, users can make informed decisions about label corrections, further enhancing the model's accuracy over time.

User participation and visualization: The "My Closet" and "Visualization" tabs are designed to increase user engagement by providing personal value beyond the classifier's primary function. Users can view and manage their labeled clothing images, save outfits, and track their wear history. These features not only serve to personalize the experience but also incentivize users to contribute to the classifier's training through active participation and feedback.

In conclusion, the proposed design leverages insights from the literature and data collection to create an interactive, user-friendly system. By emphasizing continuous learning, user feedback, and personalization, the system is not just a tool for classification but a dynamic platform that evolves with its users, ensuring relevance and improved performance over time.

1. Implementation

4.1.Administrative workflows

Administrators can efficiently manage training data by first navigating to the "Training Data" tab. Here, they update the dataset by uploading images and selecting from 10 predefined instance labels or entering a custom label. Moving on to model training, administrators access the "Training" tab to configure classifier parameters such as layers, epochs, and batchSize. Clicking "Train the classifier" initiates the training process, displaying progress via a status bar, and allowing the review of results, including loss and accuracy graphs.

In the "Model Testing" phase, administrators move to the corresponding tab, updating testing data and initiating testing with a click. Testing results, including a confusion matrix, provide valuable insights. For error analysis, administrators check the "Error reports" tab, reviewing user-audited data and confidence bar graphs. Corrections are double-checked, and accurate ones can be sent to the training set, contributing to ongoing model improvement. Technical constraints prevent seamless synchronization of datasets across tabs among users.

4.2.User workflows

Users begin by exploring the "My Closet" tab, where labeled clothing images from the "Add Clothes" tab are displayed. Saving today's outfit is a simple click, with the option to change the date or clear the outfit. In the "Adding Clothes" workflow, users navigate to the corresponding tab to add new items. The model automatically classifies items, showing a prediction confidence graph. Users can correct labels, add items to "My Closet," and optionally include custom labels. A popup window appears if label corrections were made, seeking user agreement to utilize corrected data for model training. The page can be cleared using "Clear new clothes."

For visualizing usage patterns, users explore the "Visualization" tab. Within the "Details of my clothes" section, a table summarizes the frequency of each clothing item worn, ordered by user preference. In the "Outfits history" section, users can review the chronological record of past outfits. If desired, users can clear the entire history with a click. Notably, users play a crucial role in improving the classifier by correcting labels and agreeing to use corrections for model training when prompted.

1. Evaluation method

Accurate user engagement metrics are pivotal in assessing the effectiveness of the Smart Wardrobe system interactions. By tracking key indicators such as frequency of use and interactions with labeling features, we gain valuable insights into user behavior and satisfaction levels. Higher engagement typically indicates stronger user connection and appreciation for the platform's functionality. Analyzing these metrics over time helps identify trends, areas for improvement, and ensures the system remains aligned with user needs.

Furthermore, Error analysis is vital for refining the Smart Wardrobe system by uncovering discrepancies between predicted and actual labels. Scrutinizing errors from users and administrators helps identify common patterns like misclassifications. By examining error types and causes, we pinpoint areas for improvement in the interaction design and machine learning model. Analyzing confidence bar graphs helps assess prediction certainty and prioritize corrections. By iteratively addressing errors, we enhance accuracy and improve the user experience, maximizing the system's utility.

Finally, questionnaires and narrative interviews serve as valuable tools for assessing user experience. These methods can incorporate evaluation metrics concerning various facets of user experience, including satisfaction, efficiency, ease of use, task load, trust, confidence, and effectiveness in addressing user feedback.

1. Appendices

Link to the code of implementation is as follows. <https://github.com/design-park/marcelle-smart-wardrobe>

**REFERENCES**

[1] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998d).Gradient-based learning applied to document recognition.Proceedings of the IEEE, 86(11), 22782324.

[2] Karpathy, A. (n.d.) CS231n Convolutional Neural Networks for Visual Recognition. cs231n.github.io/convolutionalnetworks.

[3] Culurciello, E. (2017, April). The history of Neural Networks.Retrieved from Dataconomy:

<http://dataconomy.com/2017/04/history-neural-networks/>.

[4] Liu Z., Luo P., Qiu S., Wang X., Tang X. (2016)DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). DOI:10.1109/CVPR.2016.124

[5] Bossard, L., Dantone, M., Leistner, C., Wengert, C., Quack,T., & Van Gool, L. (n.d.). Apparel Classification with Style.

[6] Wang, H. (n.d.). Machine Fashion: Artificial Intelligence Based Clothing Fashion Stylist.