Proposing a Sustainable AI Assistant to Reduce Environmental and Economic Cost of Fast Fashion

Wonjin Eum

Stevenson School

Abstract

Fast fashion has shifted modern consumer culture by promoting overconsumption and inconsiderate purchases, leading our team to challenge ourselves to fight against this issue. The overbearing environmental, economic, and ethical problems that fast fashion has brought to society are becoming detrimental, both individually and globally.

This AI assistant fosters sustainable consumer culture by utilizing machine learning and patterns. Incorporating the neural network system, creating a unique score function, and adopting recommendation systems, the assistant ultimately provides users with a "purchase score" that helps them minimize careless purchases.

This filtering model is generated using a deep learning code with a refined dataset. Then, the user input will be passed onto the model to get the prediction. Based on this information, the purchase score will come up via the Score function I defined. With the Score function, the Recommendation system would provide recommendation indicators by user-based and item-based systems. Finally, the optimization and personalization will present a sophisticated analysis of the consumer's purchase patterns.

The AI-assistant implements Python to build the machine learning model, and the pseudo-code and different parameters are included in the main text. The research puts emphasis on finding ways to increase the efficiency of the process and concludes with a suggestion on pattern analysis using deep learning, and vision training for the user to upload their history easily.

Introduction

A. Motivation

Modern society is strongly struggling with a wasteful consumer culture of overconsumption, especially in the fields of the fashion industry. Fast fashion is defined as inexpensive clothes produced rapidly by mass-market retailers in response to the latest trends. Some clothing brands take advantage of this and gain tremendous popularity and profit; however, fast fashion became a big global issue because of its negative impact both environmentally and ethically. The Roundup team accounts that more than 92 million tons of fast fashion textile is wasted out of the 100 billion garments purchased each year, and most of them end up in landfill [1]. Ethical Consumer also reports that the fashion industry requires 93 billion cubic meters of water and is responsible for around 20% of industrial water pollution every year [2]. Because of the endless creation of new clothes and the rapid disposal of clothes that are out of style, fast fashion creates not only a heavy environmental toll, but also a large financial price. According to Earth.Org, 500 billion dollars are lost each year because of under-wearing and failure to recycle fast fashion clothes [3]. Both the process of making fast fashion clothes and disposing it requires a lot of money; however, companies need their products to be low-priced and in-style, resulting in labor abuse and underpayment to the garment workers. Although it attracts consumers by its low-price and stylish design, fast fashion also ultimately disadvantages the consumers in a long-term perspective. If one purchases multiple fast fashion clothing, deceived by its cheapness, they will eventually spend more money compared to buying a few pricier ones that last longer. According to WBL, an average US household spends about $134 on clothing each month, primarily on fast fashion and inexpensive products that will make them purchase new clothes the next month [4]. The key to solving the fast fashion crisis is consumers starting to recognize the upsides of slow fashion and willing to become an educated shopper, as this will benefit the environment, companies, and the consumers themselves. Thus, this AI assistant was designed to change the normalized consumer culture of inconsiderate overconsumption.

B. Previous approach and limitation

A well-known consumer culture that has emerged among sustainable shoppers is slow fashion. The idea of slow fashion encourages consumers to purchase second-hand items or pricier yet stronger garments in order to increase the lifespan of an average clothing product. Slow fashion also signals people to donate and recycle clothes they already have rather than simply throwing them out. However, no matter how much slow fashion is advertised, it will never exceed the demand for fast fashion. Fast fashion's affordability and accessibility appears more attractive to consumers than expensive and basic slow fashion. The poor advertisement and low awareness of slow fashion limits the public to fully absorb the positive intention that the culture itself holds. People recycling and donating their unwanted clothes fails to fully solve the problem as well. According to The Sustainable Fashion Forum, most of the recycled clothing ultimately ends up in landfill because of the laborious process of preparing and sorting materials before recycling, difficulties in separating different materials within fiber, and lack of label content or information to identify different garment types [5]. Due to these hardships, companies need to fund a lot of money into recycling research; for example, by 2020 Inditex (Zara's parent company) invested 3.5 million dollars and H&M Foundation spend 7.2 million dollars in recycling development. While changing the consumer attitude and educating the public about the negative effects of fast fashion still remains the one and only solution to the fast fashion crisis, these previous approaches are ineffective and unapproachable. Addressing these limitations, this AI assistant will help encourage consumers to think before they purchase as well as give access to contributing in slow-fashion efforts by digitalization.

The contribution points of this paper are:

- Data collection: The primary data used for developing our AI assistant is the Fashion MNIST dataset from Kaggle. It contains a training set of 60000 examples and a test set of 10000 examples, each example being a 28x28 grayscale image. Our team is also using other clothing image CSV files from Kaggle as well for objectivity of the data.
- Data preprocessing: In order to unify the form of the different data collected, our team went through a process of data cleaning before we used it to develop the AI assistant.
- Machine learning deep learning code: Using the sorted data, our team ran the machine learning code to create an AI assistant. This process produced one of the fundamental systems of our innovation: pattern and type recognition of different garments.
- Recommendation system: The main recommendation system provides a reasonable "purchase score" to the consumer. If the consumer is trying to buy fast fashion clothing that is similar to an item they already have, the AI assistant will give them a warning. Furthermore, it will also provide the user with alternative shopping choices (under collaboration with their local thrift stores.)
- Optimization/individualization: The AI assistant presents a sophisticated analysis of the consumer's purchase patterns, giving them optimized shopping selections that are likely for them to wear a lot in the future. This individualization of data will help users understand their typical favorite garments and styles of clothes, encouraging them to make smart purchases.

The core techniques are introduced in background research (chapter II) with examples. In chapter III, the proposed system will be explained in great detail and real-life examples.

Chapter IV will explore the evaluation of the AI assistant. Lastly, the conclusion and future works will be presented in Chapter V.

Background Research

A. Neural networks

Neural networks start from neurons and their activations. The higher the activation, the more "lit up" the pixel is. Activations of one layer determine the activation of the next layer. In this example network that can learn to recognize handwritten digits, the number that has the highest activation in the last layer becomes the answer.

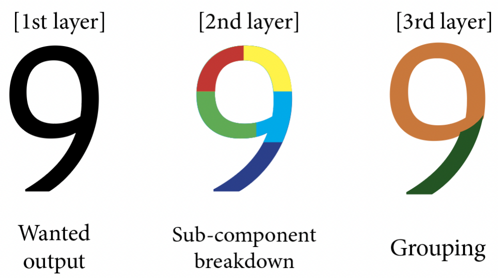

Figure 1

As shown in figure 1, each component can light up a neuron and we just have to figure out which sub-components match it. Each component corresponds to the "edge" of the number.

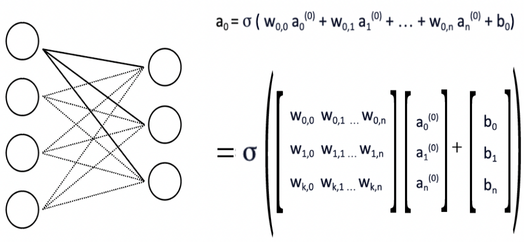

Figure 2

In order to find out if one specific region of the input is "lit up" or not, the edge detection technique is needed. Weight values are assigned to each one of the connections between the input neuron and the neurons from the first layer depending on their correspondence. Using activations and weights of each neuron, the network will then compute the weighted sum. For example, in order to detect an edge of a handwritten number digit, negative weights will be surrounding the positive weights which will allow the network to compare that to the input grid. The activation that the next layer receives becomes how positive the relevant weighted sum is. Following this logic, a bias can be set up for inactivity. If the next neuron should light up only if the weight sum is bigger than 10, that becomes a bias value.

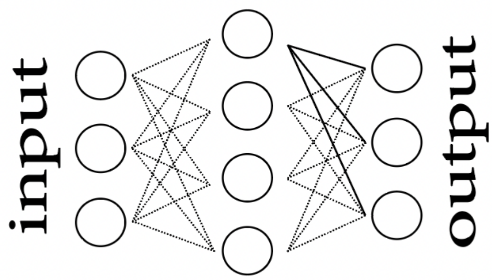

Figure 3

Ultimately, the word "learning" in machine learning refers to getting the computer to find the right weights and biases so it solves the problem. Neural networks are made up of multiple nodes, which are grouped in layers. These layers are connected to each other by weights. Each weight has a corresponding value, standing as the strength of the connection between the different layers. Once an input value comes in, the nodes will multiply its value by the weight that is connected to the next node. The weights are first randomly initialized, then it learns the optimal values for these weights, leading to a more accurate output.

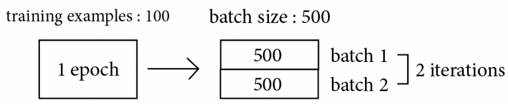

Figure 4

Epochs refers to one forward pass and one backward pass of all the training examples. In other words, it is the number of passes a training dataset takes around an algorithm. After the first epoch, the weights will set as decent. By feeding the training data to the neural network again and again, we can improve the weights further.

When one epoch is too big to feed to the computer at once, we divide it in several smaller batches. Batch size is a hyperparameter that indicates the total number of training examples present in a single batch. Depending on the epochs and the batch size, Iterations are defined as the number of batches needed to complete one epoch. For example, if we have 1000 training examples, and our batch size is 500, then it will take 2 iterations to complete 1 epoch.

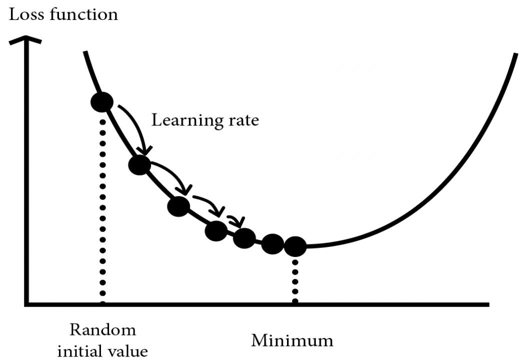

Figure 5

Loss is the error percent of the model's prediction and the actual value. Using the loss function, the optimizer will update the model in response to the output of the function to minimize the loss.

B. Recommendation System

Well-made recommendation systems are known to be both beneficial to the users and the companies. Recommendation systems use the user's past history in order to predict their future behavior, typically leading into suggesting items that the user will find attractive.

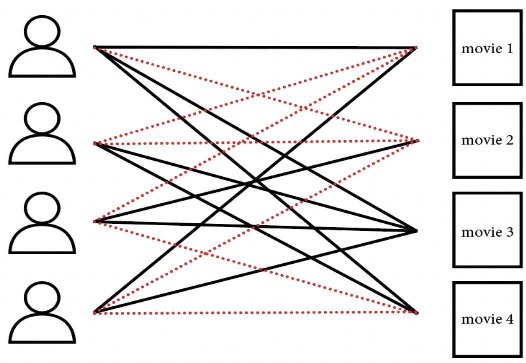

Figure 6

Matrices are used to organize user history and the predictions in one mathematical model. This example shows a matrix that is missing predictions for each individuals' preference in different movies.

	1	2	3	4
A	?	4	?	0
B	3	1	2	?
C	?	2	3	1
D	?	0	?	4

Figure 7

Content filtering is the traditional recommendation system, using information already known about the user as connective tissues for recommendations. If a user prefers a certain item, the program will recommend a similar item. First, the system will label the person and movies with different attributes and features based on a questionnaire given to the user. This will create two sets of matrices, one for the users and one for the movies, which will be multiplied and thus combined into one big matrix. The strength of the connection between and individual and a movie will be represented in this matrix, which ultimately becomes the final prediction.

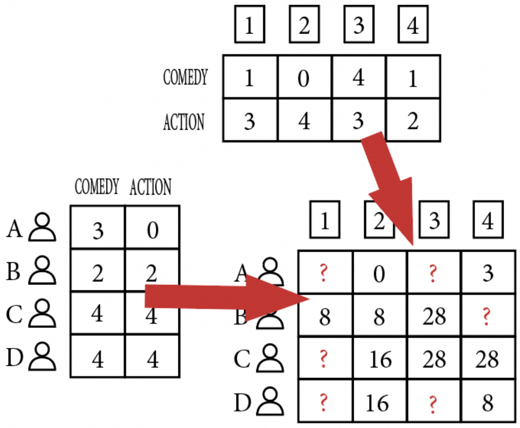

Figure 8

Collaborative filtering gives more accurate predictions as it incorporates machine learning techniques. This recommendation system gives suggestions based on things that people with similar viewing habits liked. In other words, it is a flipped version of the content filtering method. Collaborative filtering starts with a user preference data the system has and then uses machine learning to predict the two matrices for different individuals and movies. Due to this reversed-chronological structure of the method, this filtering has latent features, or average/weighted sum of the patterns in the data, that replace the features in the content filtering method.

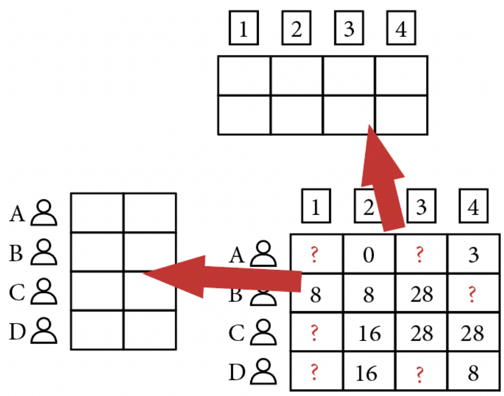

Figure 9

There are two types of collaborative filtering. The user-based method suggests that consumers similar to the user purchased certain items. For example, the identity of the user such as their age and gender become the threshold of recommendations. The item-based method suggests that consumers who purchased these items that the user liked also purchased certain items. The more the intersections of attributes between different individuals, the more likely the items one liked will be recommended to the other.

C. Optimizers

Optimizers adjusts of a neural network such as learning rates and weights in order to minimize the total loss and improve accuracy.

The most basic form of optimizers is Gradient Descent. While this stands as the basis of all the other applied optimizers, it costs a lot of expenses and errors when finding the extreme by total search through the full data.

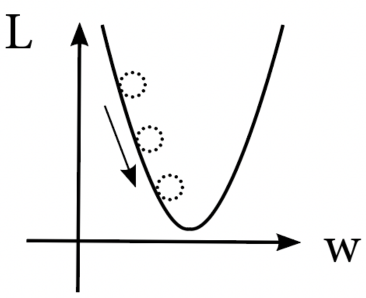

Figure 10

Another common optimizer is the Stochastic Gradient Descent (SGD). This method allows each batch to present a slightly different loss landscape, which helps with raising the accuracy.

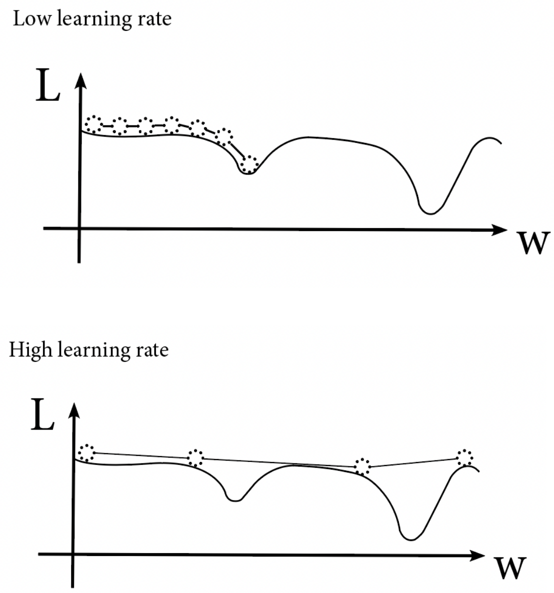

Figure 11

Similarly, SGD with momentum improves on its original prototype by characterizing the resistance of the velocity and the learning rate to influence the gradient accordingly. This makes it more adaptive to the loss landscape.

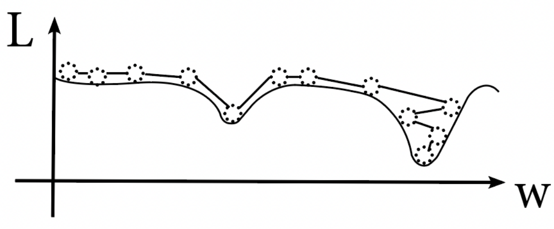

Figure 12

SGD with Nesterov momentum takes the previous optimizer one step further. The gradient is calculated at the point where W can be found after the addition of the velocity jump. If W is taken before the step, it might not fit the W after the step; by calculating the gradient at the look ahead point, it mitigates this problem and improves accuracy.

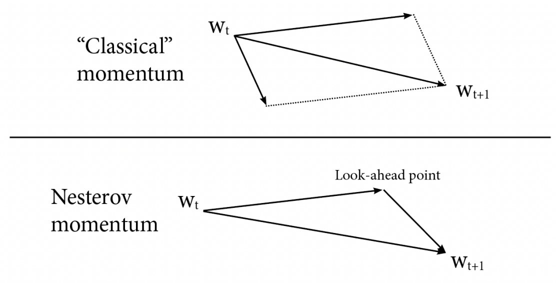

Figure 13

The adaptive gradient method, often called the AdaGrad optimizer, follows a slightly different logic than SGDs. If one parameter has changed significantly, then it must have made a lot of progress towards the target; if it didn't change much, it should be updated with greater emphasis.

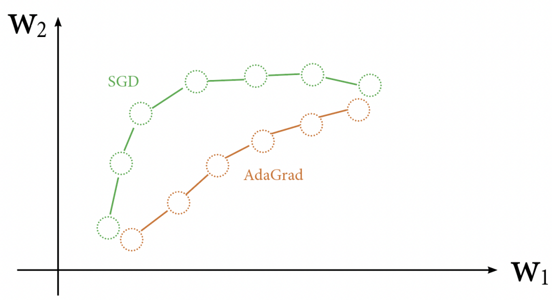

Figure 14

The RMSprop optimizer keeps some sort of memory of previous gradients as a reference point. When a large gradient is encountered, the learning rate is scaled down, and when a small gradient is encountered, the learning rate is scaled up. For example, when the surface is flat, W takes a big jump; when the surface is steep, W takes a small jump.

Lastly, Adam, short for Adaptive Moment Estimation also is used commonly despite it's complicated formulas. The gradient jump is parallel to some vector M which we can think of as an actual velocity term.

D. Loss

Loss functions quantifies the error between the output of an algorithm and a given target value. This would most commonly be made by linear regression of the actual values. The programmer's ultimate goal is to minimize the loss function through experimenting which optimizer would be the best fit.

Regression is useful when getting the value, while classification is useful when getting the category of the loss function. Within the classification method, the categorial part trains a network to output a probability over the C classes for each image, thus making it popular for multiclass problems. In contrast, the Binary part sets up a binary classification problem between C classes for every single class in C.

E. Activation

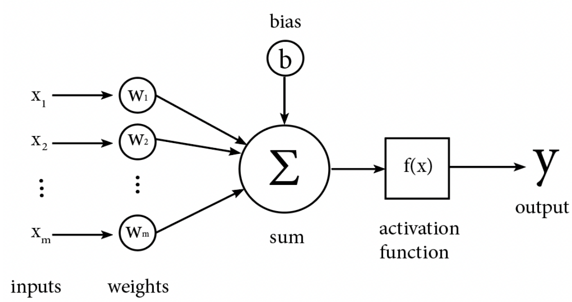

Figure 15

The activation function decides whether to fire the neuron closer to the output or not. The most basic model of the activation function is the binary (0 or 1) form. This function works when the datapoints certainly divide into two categories, but cannot identify anything other than the two.

Figure 16

Instead of binary, neurons can be activated by fractions. The below figures indicate some variants of activation functions. Each function has a distinct use that fits in with the design of the neural network; Sigmoid used for models that predict the probability, tanh used for classification, and ReLU most commonly used in general.

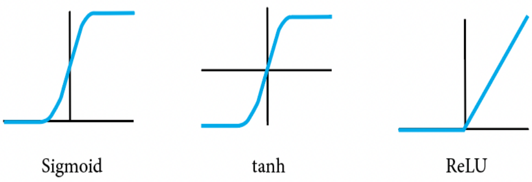

Figure 17

Proposed System

Figure N represents the whole proposed system of this paper. Specific descriptions and examples of each step are followed.

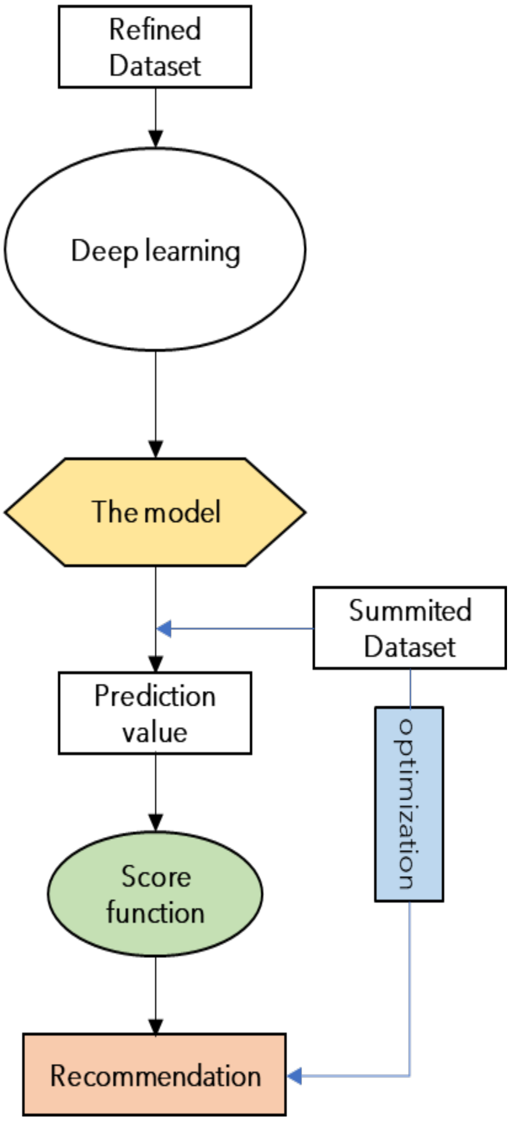

Figure 18

A. Data collection and preprocessing

This AI assistant used the Fashion MNIST dataset from Kaggle. Containing more than 60000 examples and a test set of 10000 examples, this dataset is a collection of image data commonly used for machine learning. The categories of the grayscale images include T-shirts/tops, Truser, Pullover, Dress, Coat, Shirt, Bag, and more. Each data has a dimension of 28x28 pixels, making it suitable for image classification, deep learning, and pattern recognition. The assistant also took other image CSV files from Kaggle for objectivity of the datasets. Since different data from different sources are being used, it is important to clean all the information gathered before using them to train the AI assistant. To do so, a meticulous process of data cleaning was done in order to extract a refined dataset.

B. Model Generation

The first major piece of the AI assistant is creating a model that can recognize the different patterns and types of clothes. In order to create this model, several deep learning code and the sorted dataset that was finalized in the preprocessing step. These codes ushered the computer to "learn" and find the right weights and biases to output a more accurate prediction. The details of neural networks and deep learning codes that was used can be found in chapter II, Background Research.

C. Prediction

After going through the deep/machine learning process using the training dataset, the assistant will input the real user data into the model and get a prediction out of it. This will provide an accurate analysis of the input data through pattern recognition and the machine learning knowledge that the model previously saved. For example, if a user had a flower-patterned dress in their cart, the assistant will put the image of the item into the model which will return if the user previously purchased similar patterned clothes.

D. Recommendation via Score function

The second major piece of our AI assistant is applying the model to the Score function in order to get a reasonable "purchase score" for the user. For example, once the user adds an item to their cart, the AI assistant will run through the model and the score function to output the purchase score. If the item is similar to a previous purchase in the user history, the purchase score will be very low. Making them be aware of the environmental and financial impacts this purchase will have by providing them the purchase score, the AI assistant will strongly discourage the user to buy the fast fashion item. This will serve as a friendly warning and a small amount of pressure on the user to rethink about their consumerism behavior and sustainable clothes shopping.

Score chart

The lower the score, the less beneficial it is for the user to purchase.

Factor	**Effect**	**Value (Abs, 1-10)**
similar type	neg	32
similar color	neg	24
similar pattern	neg	16
personalized?	pos	8
sustainable company	pos or neg	12
financially reasonable	pos or neg	8

Figure 19

- The word "similar" implies that the user has no reason to buy another thing like the old item; either the old one or the new one will end up thrown out because they will not be frequently worn.
- -If the item was suggested through the assistant's optimization process, the user will likely feel satisfied after the purchase.
- if the item goes over the user's budget, the score will go down, and vice versa
- if the item is sold from a sustainable company (not a fast fashion company), the score will go up, and vice versa

E. Optimization

The consistent data collection combined with the deep learning technology will allow the AI assistant to present a sophisticated analysis of the consumer's purchase patterns. In other words, optimization and personalization becomes available. For instance, if a user tends to like cotton material clothes, the AI assistant will suggest items that are made out of cotton. This will lead the users to have a higher satisfaction once they purchase it, thus increasing the value of the clothes and decreasing the possibility that it will end up in landfill after a short period of time. This individualization of data will tremendously contribute in raising awareness of the users to know their favorite garments and styles of clothes, influencing their shopping to become more sustainable and smarter.

Evaluation

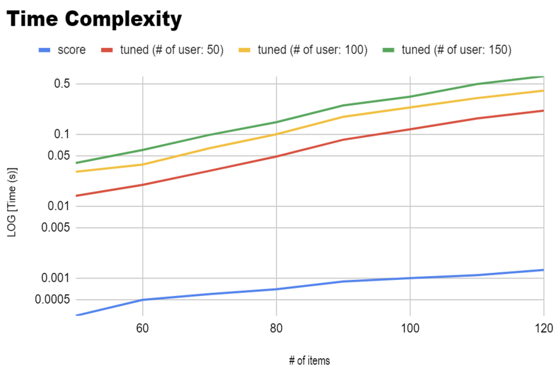

Figure 20

This logarithm scale represents the efficiency evaluation of the research. The score function calculates are linear, and applying the recommendation system makes it slightly exponential. The user and item quantity both affect the graph, but the number of users has a more significant influence. The reasonable time data shown after multiplying the quantities by 10 proves the high efficiency.

Conclusion and Future works

Fast fashion has shifted modern consumer culture by promoting overconsumption and inconsiderate purchases. This leads to environmental, economic, and ethical problems on a global scale. This research addresses this problem by using machine learning and recommendation systems to raise awareness of the seriousness of this issue.

The research puts emphasis on finding ways to increase the efficiency of the process. The results show that the final convolution neural network model successfully served in generating accurate desired outputs in the most efficient way.

In the future, I strive to build a stronger pattern analysis using deep learning and vision training for the user to upload their history more conveniently, in order to reduce the tedious process of manual input.

One challenge that would become a very interesting development in the future is redefining the recommendation system through combining user based and item-based recommendation systems, complimenting the two different codes' weaknesses and strengthening the codes' benefits

This model can be applied to real world situations through implementation in a chrome extension or a mobile app that runs behind the user's shopping browser and providing useful information for them to become an educated consumer.

Reference

1. https://theroundup.org/textile-waste-statistics/
2. https://www.ethicalconsumer.org/fashion-clothing/what-fast-fashion-why-it-problem
3. https://earth.org/statistics-about-fast-fashion-waste/
4. https://writersblocklive.com/blog/fast-fashion-statistics/
5. https://www.thesustainablefashionforum.com/pages/quick-question-why-cant-cloth
6. https://pubmed.ncbi.nlm.nih.gov/19065803/
7. https://ieeexplore.ieee.org/document/34
8. https://ieeexplore.ieee.org/document/9451544
9. https://www.riejournal.com/article_106395.html
10. https://publications.ics.forth.gr/_publications/science.pdf
11. https://arxiv.org/abs/2012.00204