Retraction

Retraction: Convoluted fashion classifier prediction using Neural Networks (J. Phys.: Conf. Ser. 1916 012029)

Published 23 February 2022

This article (and all articles in the proceedings volume relating to the same conference) has been retracted by IOP Publishing following an extensive investigation in line with the COPE guidelines. This investigation has uncovered evidence of systematic manipulation of the publication process and considerable citation manipulation.

IOP Publishing respectfully requests that readers consider all work within this volume potentially unreliable, as the volume has not been through a credible peer review process.

IOP Publishing regrets that our usual quality checks did not identify these issues before publication, and have since put additional measures in place to try to prevent these issues from reoccurring. IOP Publishing wishes to credit anonymous whistleblowers and the Problematic Paper Screener [1] for bringing some of the above issues to our attention, prompting us to investigate further.

[1] Cabanac G, Labbé C and Magazinov A 2021 arXiv:2107.06751v1

Retraction published: 23 February 2022
Convoluted fashion classifier prediction using Neural Networks

T Rajesh Kumar 1, S Kishor 1, R U Kishore 1, C Naveen Kumar 1
1 Department of Information Technology, Sri Krishna College of Technology, Coimbatore, India
17tuit036@skct.edu.in

Abstract. The conventional way of choosing a fashion catalogue has a lot of drawbacks and is yet to be resolved. Struck in a dilemma of what to choose and what not to, not wearing fashion items that are appropriate, and lured by terms that are misleading such as 50 percent off limited period offer, when in reality the prize was never higher in the first place. These are the flaws that go under the radar for humans, but that is not the case with machines. The proposed system when given an image of a fashion item, by the user makes use of a Convolution Neural Network to find the type of item, then suggest items based on occasion. The system also gives you the prize of that particular item over a period of time in various e-commerce sites, and if there is any chance of drop in prize in the future. This eliminates the human part of errors while choosing a fashion catalogue and makes everything automated. The next advantage of our site is to resell our old clothes at best prize so our used clothes are not wasted and it can be used by others.

keywords: Fashion, CNN, web-scraping, Django Rest Framework, ReactJS

1. Introduction
The fashion world has enhanced with lots of varieties of products and online fashion store has been used by so many people. In these covid situation people do not go outside to purchase items such as clothes, shoes, make up items [1]. There is a confusion on which to choose from lots of products so to make shopping easier we have proposed a system on which a product matches the other. For instance a shirt is chosen the best matching bottom would be recommended by CNN [2]. The system also tells us on which day the product would be in the best price and it also tells the upcoming offers in the store so you can save money.

The best price is predicted by collecting the data of previous price of around 10 days and our system tells the best price of a product in upcoming days. It is built by bs4 module in python [3] for accuracy and it keeps track of all items over a period to find the best price.

Another feature in our system is we can sell and buy our reused clothes at best price. We have used MNIST dataset [4] to train and validate our system. The reused clothes is only considered in our website if it is in a good condition and branded by doing this we can reuse our clothes and no fashion items[5] are of no use. The system will provide the best price for our cloth and other items.

The idea of our system is normal store does not tell us when to buy the product at best price and there is no category for selling reused clothes by doing this many people could save money and time. Many users would use our website since it makes shopping easy and it saves our money as well. Introduction
2. Convolution Neural Network based fashion classifier

A convolution neural network is a deep learning algorithm that uses the convolution operation and the back propagation algorithm to identify and classify a given image. We can separate any CNN classifier into three sections i.e. Convolution, max pooling [6] and fully connected network.

The Convolution layer is where the convolution operation is applied on the images, after every epoch the neurons in the layers are tuned to find features from the image. The first layer would be tuned to find vertical lines the second would identify shapes and as we go deeper and deeper through the layers the features would become more complex in nature. The architecture of CNN is depicted below in Figure 1.

![CNN Architecture Diagram](image-url)

**Figure 1.** CNN architecture

Max pooling [7] is effectively for reducing the number of computations by reducing the number of pixels, that is we pool out 4 pixels (varies based on value passed) as one, effectively reducing the image by 75%. Fully connected network is where the 2D feature set is converted to a linear value and then classified into various categories.

Now in our proposed system we have used the concept of transfer learning[8], which is basically using the patterns and knowledge gained while solving one particular problem to solve another problem which is related. For example a model trained to detect trucks can be fine tuned slightly to detect cars. This reduces the number of computations one has to do and reduces cost.

Alexnet’s contains a total of 60000 parameters with 8 total layers five convolutional layers, three fully connected layers. Alexnet[9] uses Rectified linear unit (ReLu) as the activation function which reduces the error and makes it faster. A sample of Fashion-MNIST Dataset is given below in Figure 2.

Retracted
Figure 2. Fashion-MNIST Dataset

We have used the fashion-MNIST dataset done by Zalando Research\cite{10}, containing a total number of 70,000 (60,000+10,000) training and test validation grayscale images, where each image is labelled as one of ten types of clothing (such as t shirt, coat, sneaker, etc.). The distribution of this dataset is summarized in Figure 3 below.

| Code | Label       | Training | Validation | Testing |
|------|-------------|----------|------------|---------|
| 0    | T-shirt/top | 6,000    | 1,000      | 232     |
| 1    | Trouser     | 6,000    | 1,000      | 404     |
| 2    | Pullover    | 6,000    | 1,000      | 431     |
| 3    | Dress       | 6,000    | 1,000      | 324     |
| 4    | Coat        | 6,000    | 1,000      | 430     |
| 5    | Sandal      | 6,000    | 1,000      | 201     |
| 6    | Shirt       | 6,000    | 1,000      | 461     |
| 7    | Sneaker     | 6,000    | 1,000      | 274     |
| 8    | Bag         | 6,000    | 1,000      | 294     |
| 9    | Ankle boot  | 6,000    | 1,000      | 297     |
|      | Total       | 60,000   | 10,000     | 3,348   |

Figure 3. Training, Validation and testing data Distribution Table

Using this model we will classify the images into fashion categories then based on the occasion we will recommend clothing material to be worn with the given outfit if the product id is specified we can track the product's prize over a period of time and give details like maximum prize it was sold for and a graph on prize trends over a period of time.

Retracted
Along with this, users can sell their old clothes, and the network will give an approximate prize after analyzing based on the product ID and how long it has been used. The summary of AlexNet parameters is given below Figure 4.

![Figure 4. Summary of AlexNet Parameters](image)

The proposed system uses the Keras framework for building the neural network model. The model architecture used is LeNet-5 [11] for more accuracy. The system makes use of the bs4 module in Python to web scrape the product's price over a period of time based on the product's ID. The system also makes use of the ReactJS and Django rest framework [12] to build the web interface.

3. Implementation and result:

We were able to build our neural network model using the Keras framework in Python, we trained our neural network model using Google Colab which is an online collaborative tool where we can make use of Google’s fast GPU power [13] to train our model. After selecting the model structure, the key problem was to select training variables [14] like learning rate, number of epochs, and batch size. We tried different values and based on graph simulations and accuracy, we locked our training variables. We then built an web-based interface where users can upload their fashion items and then get the required functionality [15], we did not want the users to use the command line interface. The interface was built using the Django rest framework and ReactJS library.
Given an image of any fashion item the system identifies the type of image using a neural network and then suggests items that suits the given item and items based on the occasion [16]. The proposed system includes a price tracker, which keeps track of an item's price over a period of time based on the product id. This helps in avoiding falling for discount traps set by many e-commerce websites [17]. The proposed system also allows users to sell their used fashion items [18]. The system gives out an appropriate price based on date of purchase and the price of the product when it was purchased. Figure 5 depicts the price vs time graph of an amazon product [19].

![Price vs time graph of an amazon product](image)

**Figure 5.** Price of dress is predicted

Loss over Epochs while training our neural network is depicted below in Figure 6.
4. Conclusion and future enhancement

There are definitely a few areas where we can improve. We can also train the model to give the perfect size that would fit the person, improve the accuracy of the current model and venture out to a greater number of fashion products. We can also add a feature called try this, where users can have a visual 3-dimensional image of them wearing that particular fashion item.

References

[1] Hu, W., Huang, Y., Wei, L., Zhang, F., & Li, H. (2015). Deep convolutional neural networks for hyper spectral image classification. Journal of Sensors, 2015.
[2] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Image Net classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).
[3] Najafabadi, M. M., Villanustre, F., Khoshgoftaar, T. M., Seliya, N., Wald, R., & Muharemagic, E. (2015). Deep learning applications and challenges in big data analytics. Journal of Big Data, 2(1), 1.
[4] Chen, X. W., & Lin, X. (2014). Big data deep learning: challenges and perspectives. IEEE access, 2, 514-525.
[5] Noda, K., Yamaguchi, Y., Nakadai, K., Okuno, H. G., & Ogata, T. (2015). Audio-visual speech recognition using deep learning. Applied Intelligence, 42(4), 722-737.
[6] Jianming, Z. H. A. N. G., Zhicai, Z. H. A. N., Keyang, C. H. E. N. G., & Yongzhao, Z. (2015). Review on development of deep learning. Journal of Jiangsu university: natural science editions, 36(2), 191-200.
[7] Yuan, A., Bai, G., Jiao, L., & Liu, Y. (2012, March). Offline handwritten English character recognition based on convolutional neural network. In 2012 10th IAPR Madhusudhan M and Shankaraiah, “Implementation of automated door unlocking and security system,” International Journal of Computer Applications, pp. 5-8, 2015.

[8] Kaggle, (2019), https://www.kaggle.com/zalando research/fashionmnist, April, 2019.

[9] Hu, W., Huang, Y., Wei, L., Zhang, F., & Li, H. (2015). Deep convolutional neural networks for hyperspectral image classification. Journal of Sensors, 2015.

[10] Alom, M. Z., Taha, T. M., Yakopovic, C., Westberg, S., Sidike, P., Nasrin, M. S., & Asari, V. K. (2019). A state-of-the-art survey on deep learning theory and architectures. Electronics, 8(3), 292.

[11] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

[12] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

[13] D. Devikanniga, A. Ramu, and A. Haldorai, Efficient Diagnosis of Liver Disease using Support Vector Machine Optimized with Crows Search Algorithm, EAI Endorsed Transactions on Energy Web, p. 164177, Jul. 2018. doi:10.4108/eai.13-7-2018.164177

[14] H. Anandakumar and K. Umamaheswari, Supervised machine learning techniques in cognitive radio networks during cooperative spectrum handovers, Cluster Computing, vol. 20, no. 2, pp. 1505–1515, Mar. 2017.

[15] Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S. (2019). A survey of the recent architectures of deep convolutional neural networks. arXiv preprint arXiv:1901.06032.

[16] Rajesh kumar,T., Remmiya devi,G., Abinaya,K., Deepika ,N N., Priyadharshini,S., An Integrated Density Based Traffic Load Balancing System in a Cloud Environment, Pak. J. Biotechnol. Vol. 14 (4) 623-627 (2017), ISSN Print: 1812-1837, ISSN Online: 2312-7791

[17] Rajesh Kumar Thangavel,. et.al, Blackspot Alert and Accident Prevention System2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), IEEE Xplore, 10.1109/ICCCNT45670.2019.8944412

[18] Rajesh Kumar,T., et.al, Process Efficient Artificial Neural Network-Based Approach for Channel Selection and Classification of Seizures, Computational Network Application Tools for Performance Management, Asset Analytics, doi.org/10.1007/978-981-32-9585-8_12

[19] Rajesh kumar,T Samsudeen, S. Sangeetha, S. Sudha Rani, Enhanced Visual Attendance System by Face Recognition USING K-Nearest Neighbor Algorithm Journal of Advanced Research in Dynamical & Control Systems, Vol. 11, 06-Special Issue, 2019