FASHION RECOMMENDATİON SYSTEMS

Adressed Problem

Fashion recommendation systems offer several benefits for both businesses and consumers. Here are some key advantages:

Personalized Shopping Experience: Fashion recommendation systems provide users with personalized product suggestions based on their preferences, past purchase history, and browsing behavior. This enhances the overall shopping experience by showing users items that align with their individual styles.

Increased User Engagement: By offering relevant and personalized recommendations, these systems can increase user engagement on e-commerce platforms. Users are more likely to spend time exploring and discovering new products when the recommendations are tailored to their tastes.

Enhanced Customer Satisfaction: When users find items that match their preferences easily, it leads to higher customer satisfaction. Fashion recommendation systems contribute to creating a seamless and enjoyable shopping journey, increasing the likelihood of repeat business and positive reviews.

Cross-Selling and Upselling Opportunities: These systems can suggest complementary or higher-end items, leading to cross-selling and upselling opportunities. By showcasing related products, businesses can increase the average transaction value and maximize revenue.

Improved Inventory Management: Fashion retailers can optimize inventory levels and reduce waste by leveraging recommendation systems. By promoting products that are likely to be popular among customers, retailers can better manage their stock and avoid overstocking or understocking issues.

Data-Driven Insights: Fashion recommendation systems generate valuable data on customer preferences, behavior, and trends. Retailers can leverage this data to gain insights into market trends, identify popular styles, and make informed decisions about inventory, marketing strategies, and product development.

Competitive Advantage: Offering personalized recommendations provides a competitive edge in the crowded e-commerce landscape. Businesses that deploy effective fashion recommendation systems can differentiate themselves, attract more customers, and retain a loyal user base.

Time Savings for Users: Users can save time by quickly discovering products that align with their preferences, rather than manually searching through a large inventory. This convenience is particularly important in the fast-paced world of fashion.

Adaptability to Changing Trends: Fashion recommendation systems can adapt to changing trends and user preferences. By continuously analyzing data, these systems can stay current with the latest fashion trends, ensuring that recommendations remain relevant.

Enhanced Marketing Effectiveness: Businesses can use recommendation systems to create targeted marketing campaigns. By understanding individual preferences, retailers can deliver personalized promotions and advertisements, increasing the likelihood of conversion.

A literature revive and a research is done on fashion recommendation systems in the first week. In addition to this a template is formed for our problem. In the second week , a CNN algoritm and its testing is done. The CNN mentioned in the literature review is thought to be used in the design of recommendation system. However, no recommendation system is done. The study is conducted on Fashion Product Images (small) kaggle dataset.[1]

Jupiter notebook has one of the most preferred used deep learning libraries. We will use it to create our firts CNN. We will use the Fashion Product Images (small) kaggle dataset. The dataset has 44000 products with category labels and images. The 70 % of data will be used to train the model, 30% of the data will be used to test the data.

In the 4 th week, we studied on the first of these notebooks. In this notebook we used VGG16. VGG16 is a specific CNN architecture developed by the Visual Geometry Group at the University of Oxford. It was introduced as part of the ImageNet Large Scale Visual Recognition Challenge in 2014. VGG16 is known for its simplicity and effectiveness in image classification tasks.

Key Components:

1. **Architecture:** VGG16 has a relatively simple and uniform architecture, consisting of 16 layers (13 convolutional layers and 3 fully connected layers).
2. **Filter Size:** VGG16 uses small 3x3 convolutional filters with a stride of 1 in each convolutional layer, creating a deep network with a small receptive field.
3. **Pooling:** Max-pooling layers are used to downsample the spatial dimensions and reduce the number of parameters.

Use Cases: VGG16 is often used as a feature extractor or a pre-trained model for various computer vision tasks. Researchers and practitioners may use transfer learning, fine-tuning VGG16 on a specific dataset, to leverage its learned features for new applications.

In summary, CNNs, in general, are a class of neural networks designed for processing grid-like data, and VGG16 is a specific CNN architecture known for its simplicity and effectiveness in image classification tasks.

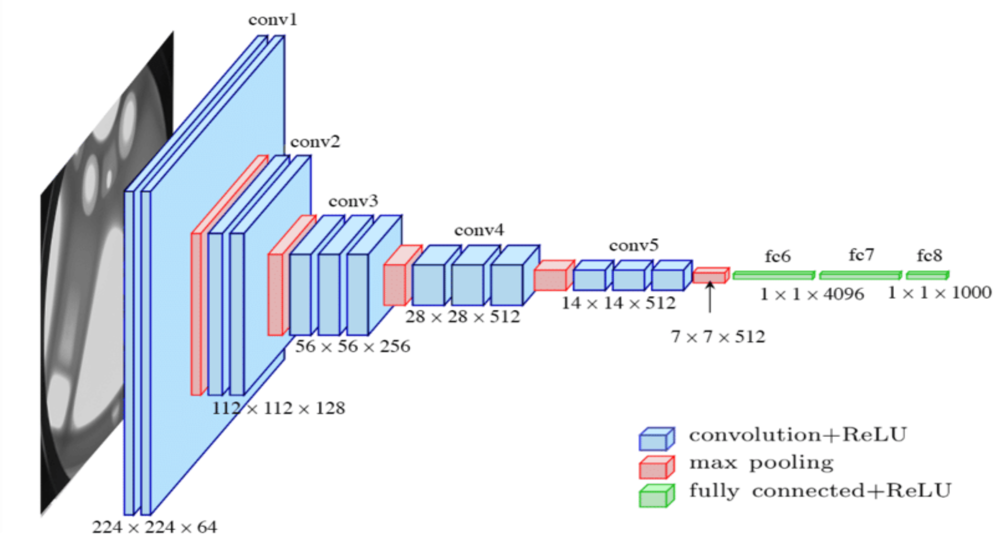

Figure 1[2]

On the last week, we considered Content based filtering based on the description of an item. Unlike the similarity calculation with cosine similarity, in this system we used KNN. KNN stands for k-Nearest Neighbors, and it is a simple, non-parametric, and supervised machine learning algorithm used for classification and regression tasks. It is a type of instance-based learning, where the algorithm makes predictions based on the majority class (for classification) or average value (for regression) of the k-nearest data points in the feature space.

Here's how the KNN algorithm works:

1. **Training:** The algorithm memorizes the entire training dataset.
2. **Prediction (Classification):** For a new input data point, the algorithm identifies the k-nearest neighbors from the training dataset based on a distance metric (commonly Euclidean distance). The class of the majority of these neighbors becomes the predicted class for the new data point.
3. **Prediction (Regression):** For regression tasks, the algorithm calculates the average of the k-nearest neighbors' target values and assigns it as the predicted value for the new data point.

KNN is often used as a baseline algorithm for comparison with more complex models. It is suitable for smaller datasets and can be effective in scenarios where decision boundaries are non-linear and complex.

REFERENCES

[1] https://www.kaggle.com/datasets/paramaggarwal/fashion-product-images-small

[2] https://medium.com/@ommore524/vgg-16-convolution-neural-network-bae747a7494a