INTERNATIONAL JOURNAL · ENGINEERING RESEARCH

International Journal for Innovative Engineering and Management Research

PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL

www.ijiemr.org

IJIEMR Transactions, online available on 05th Apr 2023. Link :http://www.ijiemr.org/downloads.php?vol=Volume-12&issue=Issue 04

10.48047/IJIEMR/V12/ISSUE 04/16

Title IMAGE AND TEXT BASE PRODUCT RECOMMENDATION

Volume 12, ISSUE 04, Pages: 113-122

Paper Authors

CH. Hari Prasad, Syed Shaheer ,Shaik Heena Fathima, Shaik Mohammed Jaseem, Vuyyuru Tejaswini

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

Image and Text Base Product Recommendation

CH. Hari Prasad[1], Assistant Professor, Department of CSE,
Vasireddy Venkatadri Institute of Technology, Nambur, Guntur Dt., Andhra Pradesh.

Syed Shaheer[2] **,Shaik Heena Fathima**[3], **Shaik Mohammed Jaseem**[4], **Vuyyuru Tejaswini**[5]

[2,3,4,5] UG Students, Department of CSE,
Vasireddy Venkatadri Institute of Technology, Nambur, Guntur Dt., Andhra Pradesh.
hari.chandika@gmail.com, sdshaheer5421@gmail.com, heenafathima2811@gmail.com, shaikmohammedjaseem656@gmail.com, tejaswinivuyyuru.1910@gmail.com

Abstract

The explosion of e-commerce platforms in recent years has brought about an enormous volume of product data. Recommender systems have become increasingly important in the e-commerce industry to help users navigate through this vast amount of data and to provide personalized recommendations based on user preferences. Traditionally, recommendation systems have relied on either image-based or text-based features to make recommendations. However, combining these two features could potentially lead to more accurate and effective recommendations. In this paper, we propose an approach that combines image and text-based features to provide more accurate and personalized product recommendations. We use ResNet-50 to extract image embeddings, and the Sentence-Transformers model with the BERT-base-NLI-mean-tokens architecture to generate text embeddings. Cosine similarity is then used to measure the similarity between the embeddings, which serves as the basis for product recommendations. The main contribution of this paper is to navigate the effectiveness of combining image and text-based features for product recommendations. Specifically, we evaluate the proposed approach on a large dataset of product images and descriptions to get recommendations. We also focus on computability to make our approach run on commodity-level hardware with a single GPU.

Keywords: BERT, E-commerce, Embeddings, Image-based Recommendation, Text-based Recommendation, Product Recommendation, Recommender Systems, ResNet-50, Sentence-Transformer, Cosine Similarity, Personalized Recommendations.

Introduction

Recommender systems are algorithms designed to provide personalized recommendations to users, based on their historical interactions with a system or other users with similar interests. These systems are used in a variety of domains, including e-commerce, social networks, streaming services, and more.

One example of a recommender system is Amazon's product recommendation system. Amazon collects data on users' browsing and purchasing history, and uses this data to suggest products that they might be interested in. Another example is Netflix's movie recommendation system, which uses a user's viewing history and ratings to suggest movies and TV shows that they might enjoy.

Recommender systems have many uses, including increasing sales and customer engagement, improving user satisfaction and loyalty, and reducing search time. They can also help users discover new products or content that they might not have otherwise found.

However, there are also some drawbacks to recommender systems. One common issue is the "filter bubble" effect, where users are only exposed to recommendations that are similar to what they have previously interacted with, potentially limiting their exposure to diverse content. Additionally, there can be privacy concerns related to the collection and use of user data for personalized recommendations. Finally, it can be challenging to evaluate the effectiveness of recommender systems, as the metrics used to measure their success can be subjective or difficult to interpret.

Our method is based on the feature extraction capability of CNN's [1] and Transformers [2]. Convolutional Neural Network (CNN) is a type of deep neural network that is particularly effective at image processing and computer vision tasks. One of the key strengths of CNNs is their ability to extract useful features from images. The breakthrough of AlexNet in 2012 [3] marked a turning point in the development of Convolutional Neural Networks (CNNs) for image processing and computer vision tasks.

Literature Survey

Our method is based on the work done in [4]. The paper proposes an image-based content recommendation system using a pre-trained deep learning model and specifically, a convolutional neural network (CNN) to extract features from images. The model generates feature vectors for each image and calculates the cosine similarity between them to recommend similar images. The VGG-16 architecture is used for classification and a linear base model Support Vector Machine (SVM) is associated with it. The proposed method is built on features from users' chosen images and suggesting similar images based on visual similarity. The cosine distance metric is used to compute the similarity score between feature vectors, and the model achieves good results using this approach.

We used a larger Fashion dataset that is 2.5x larger (5000 images) than the one used in the base paper (2000 images). The dataset is from Kaggle, which is openly available. Next, we used ResNet-50 [5] and the base paper used VGG [6] as its feature extractor. The major difference that drastically improved our recommendations is that we used the textual metadata of a product to get text-based recommendations. These text embeddings helped us in getting the most similar recommendations by focusing on various local attributes of the product whereas image embeddings achieved diverse predictions by focusing on global attributes.

Problem Identification

There are various ways to build a recommendation system. Whatever the method is, the final goal stays the same: Given a query product, predict the most likely products that might be liked by the user viewing the query product.

Collaborative filtering

This method [8] recommends items to a user based on the preferences and behaviors of similar users. Collaborative filtering can be done using user-based or item-based approaches. Also, this is the method behind Netflix's recommendation system.

Content-based filtering

This method [9] recommends items to a user based on the characteristics of items that the user has previously shown interest in. For example, a content-based movie recommendation system might recommend other movies with similar genres or actors.

Hybrid recommender systems

These systems [10] [14] combine multiple recommendation methods to provide more accurate and diverse recommendations. For example, a hybrid system might use collaborative filtering and content-based filtering together.

Matrix factorization

This method [11] breaks down a user-item interaction matrix into two lower-dimensional matrices representing user and item features, which can then be used to make personalized recommendations.

Association rule mining

This method [12] finds associations between items that frequently occur together in users' transaction histories, and uses those associations to make recommendations.

Deep learning-based recommendation systems

These systems [13] [15] use deep neural networks to learn patterns and relationships in user-item interactions and make personalized recommendations.

Proposed Methodology

In our methodology, we aimed to create a simple and straightforward approach to develop an image-based content recommendation system. We used well-established pre-trained models, ResNet-50 for image embeddings and Sentence-Transformer for text embeddings, to extract features from the images and textual data, respectively. By using these pre-trained models, we avoided the need for extensive training on large datasets, which can be computationally expensive and time-consuming [17].

Furthermore, we used cosine similarity [18] as our metric for recommendations, which is a widely used and effective technique for measuring similarity between vectors. This simplified the process of recommendation as it does not require complex algorithms or machine learning models to be built from scratch.

A. Methodology

Using pre-trained CNNs for this problem has several advantages over training from scratch. Pre-trained CNNs have been trained on large datasets such as ImageNet [19] with millions of images, allowing them to learn general features and patterns that are useful for a wide range of tasks, including image classification and feature extraction. By using pre-trained CNNs, we can leverage the knowledge gained from training on such a large dataset and transfer it to our own task with relatively little additional training data.

Moreover, pre-trained CNNs often have a large number of parameters that need to be optimized during training, and training from scratch can be computationally expensive and time-consuming. Using pre-trained CNNs allows us to avoid this computational burden and save significant time and resources.

ResNet-50, in particular, has been shown to be a powerful pre-trained CNN for a wide range of image-based tasks. It consists of 50 layers and has residual connections, which enable the network to learn more complex representations of the input images. ResNet-50 has been trained on the large-scale ImageNet dataset and has been shown to generalize well to a wide range of images outside of the ImageNet dataset. This means that the features learned by ResNet-50 can be useful for a wide range of image-based tasks beyond those it was specifically trained on.

Overall, using pre-trained CNNs such as ResNet-50 for image embeddings can provide significant advantages in terms of performance, computational efficiency, and generalization to a wide range of images.

Sentence Transformer is a pre-trained model that has been specifically designed to produce sentence-level embeddings. These embeddings are vectors that capture the semantic and syntactic information of a sentence, making them ideal for downstream natural language processing (NLP) tasks such as text classification, information retrieval, and sentiment analysis.

Sentence Transformer makes it easy to obtain high-quality sentence embeddings without requiring extensive training data or complex neural network architectures. Instead, it leverages pre-existing transformer models, which have been shown to be highly effective at natural language processing tasks.

One of the main advantages of using Sentence Transformer is that it allows for transfer learning. This means that the model can be pre-trained on large amounts of text data and then fine-tuned on specific tasks, such as text classification or information retrieval, with relatively small amounts of task-specific data. This can greatly reduce the amount of training time and data required, making it easier and faster to develop high-performing NLP models.

Cosine similarity is a measure of similarity between two non-zero vectors of an inner product space. In the context of this paper, it is used to measure the similarity between the image and text embeddings obtained from ResNet-50 and Sentence-Transformer, respectively. The cosine similarity score ranges from -1 to 1, where a score of 1 indicates that the two vectors are identical, and a score of -1 indicates that they are opposite. The closer the cosine similarity score is to 1, the more similar the two vectors are considered to be.

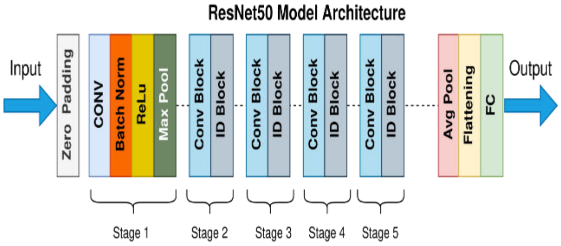
In the proposed methodology, the cosine similarity score is used to predict the relevance of images to a given text query. Specifically, the text query is first passed through Sentence-Transformer to obtain its embedding, which is then compared with the embeddings of all images using cosine similarity. The images with the highest cosine similarity scores are considered to be the most relevant to the text query and are recommended to the user.

B. Architecture

The architecture of the proposed method for image-based content recommendation system consists of two main components:

1. Image Encoder (Image Feature Extractor): In this component, a pre-trained ResNet-50 model is used to extract image embeddings from input images. ResNet-50 is a deep convolutional neural network architecture that has been trained on millions of images from the ImageNet dataset. The pre-trained ResNet-50 model is used as a feature extractor, and its last fully connected layer is removed to obtain a 2048-dimensional feature vector for each input image.
2. Text Encoder: In this component, a pre-trained sentence transformer model is used to encode text descriptions of the images into fixed-dimensional vector representations. The sentence transformer model takes textual input, processes it through a series of transformer layers, and outputs a dense 768-dimensional vector representation of the input text.

After obtaining image embeddings and text embeddings, cosine similarity is used to compute the similarity scores between the input image and all other images in the dataset. The images with the highest similarity scores are then recommended to the user.



ResNet-50 Architecture

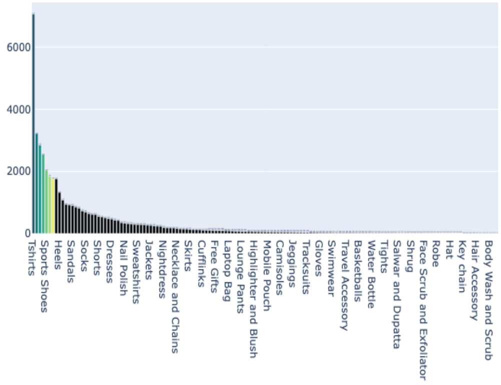
Using image embeddings and text embeddings helped in achieving the results mentioned above because they encode different aspects of the products. Image embeddings capture the visual characteristics of the product, such as the style, shape, texture, and patterns, while text embeddings capture the textual characteristics of the product, such as the description, title, and brand name.

By combining both image embeddings and text embeddings, the model was able to make more accurate and diverse recommendations. For example, if a user liked a red shirt with a certain style, the image embeddings would recommend similar products with the same style, but not necessarily the same color. On the other hand, the text embeddings would recommend products with the same style and the same color, making the recommendations more precise and relevant.

Implementation

The code is implemented in Python 3 in a Kaggle kernel. Using Kaggle kernel has eased our work a lot by reducing the time to load and download the dataset everytime into a new notebook session. Furthermore, Kaggle provide Nvidia's P100 GPU with 16GB memory. This is better than the Google Colab's GPU which is provided on a per-demand basis. We used Tensorflow for all model implementations and predictions.

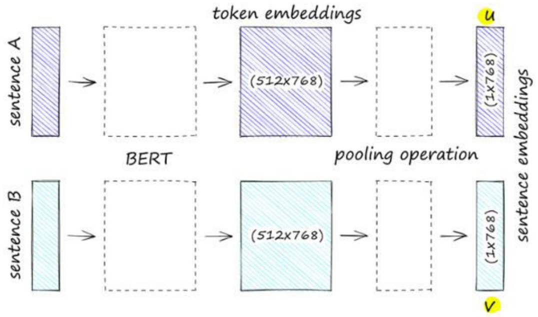
We first made a basic EDA that helped in getting important information about the dataset.



Product Category Distribution

Next, we started on getting image embeddings using a ResNet-50 model pretrained on ImageNet dataset. Given a query image, we calculated the cosine similarity between the query image and all other images' emeddings. Then finally sorted the similarity scores in descending order and we took the top-5 similar products as our recommendations.

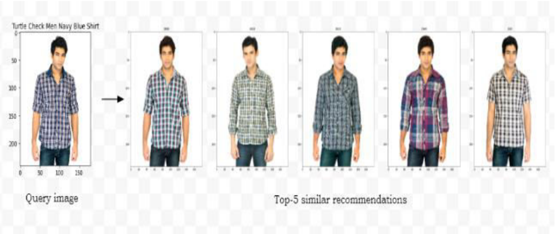
The same procedure is followed for text based recommendation. We first concatenated the textual data given in the dataset (category, year, season, product name, etc.). Then passed each joined sentence through a sentence transformer, this way we obtained sentence embeddings for all sentences. For a given query product, we first calculated the cosine similarity between sentence embeddings of the query image and all the images in the dataset. Then we took the top-5 similar products as recommendations.

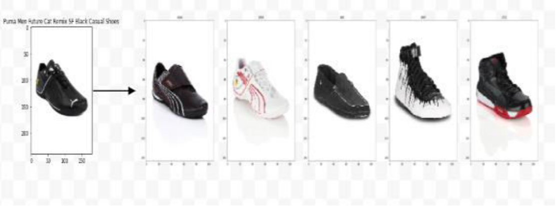


An SBERT model applied to a sentence pair *sentence A* and *sentence B*. Note that the BERT model outputs token embeddings (consisting of 512 768-dimensional vectors). We then compress that data into a single 768-dimensional sentence vector using a pooling function.

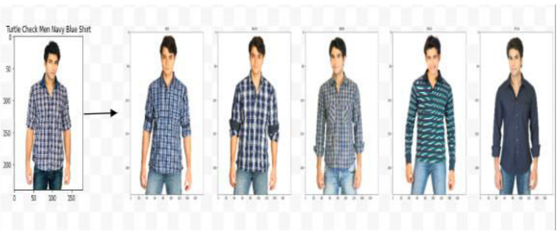
Results & Conclusion

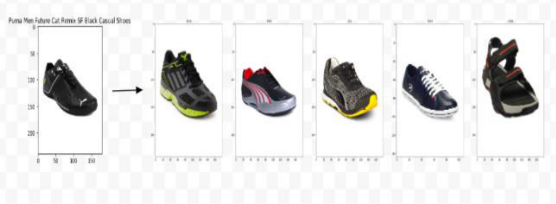
The results using image embeddings were of same style but not same colour. We can see it in the below figure:


Turtle Check Men Navy Blue Shirt
Query image
Top-5 similar recommendations



The results using textual embeddings were able to focus on recommending products of similar colours:





Limitations & Future Scope

Although the proposed methodology using ResNet-50 for image embeddings and Sentence-Transformer for text embeddings achieved promising results, there are still limitations and areas of improvement that can be explored in future research.

One limitation is that ResNet-50 was trained on ImageNet, which is a dataset that contains a wide variety of images, but it still may not capture all the nuances of fashion product images. Using more powerful models like OpenAI's CLIP [20] (Contrastive Language-Image Pre-Training) that has been trained on a larger and more diverse dataset like COCO [21] (Common Objects in Context) can lead to better image embeddings and, in turn, better recommendations.

Another limitation is that the current method only uses cosine similarity to calculate the similarity between image and text embeddings. More advanced techniques like the COLA [22] (Cross-modal and Language-based Retrieval) architecture can be used to combine both image and text embeddings to obtain a more accurate measure of similarity. The COLA architecture uses a multi-modal encoder that can encode both images and text and a cross-modal retrieval module that can retrieve the most relevant images and text given a query.

References

[1] M. Jogin, Mohana, M. S. Madhulika, G. D. Divya, R. K. Meghana and S. Apoorva, "Feature Extraction using Convolution Neural Networks (CNN) and Deep Learning," 2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, India, 2018, pp. 2319-2323, doi:10.1109/RTEICT42901.2018.901250.

[2] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17). Curran Associates Inc., Red Hook, NY, USA, 6000–6010.

[3] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet classification with deep convolutional neural networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1 (NIPS'12). Curran Associates Inc., Red Hook, NY, USA, 1097–1105.

[4] A. Angadi, S. Keerthi Gorripati, V. Rachapudi, Y. Krishna Kuppili and P. Dileep, "Image-based Content Recommendation System with CNN," 2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 2021, pp. 1260-1264, doi:10.1109/ISMAC52330.2021.9641026.

[5] *He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. ArXiv. https://doi.org/10.48550/arXiv.1512.03385*

[6] S. Liu and W. Deng, "Very deep convolutional neural network based image classification using small training sample size," 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR), Kuala Lumpur, Malaysia, 2015, pp. 730-734, doi: 10.1109/ACPR.2015.7486599.

[7] *Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. ArXiv. https://doi.org/10.48550/arXiv.1908.10084*

[8] R. Zhang, Q. -d. Liu, Chun-Gui, J. -X. Wei and Huiyi-Ma, "Collaborative Filtering for Recommender Systems," 2014 Second International Conference on Advanced

Cloud and Big Data, Huangshan, China, 2014, pp. 301-308, doi: 10.1109/CBD.2014.47.

[9] Zisopoulos, Charilaos & Karagiannidis, Savvas & Demirtsoglou, Georgios & Antaris, Stefanos. (2008). Content-Based Recommendation Systems. https://www.researchgate.net/publication/236895069_ContentBased_Recommendation_Systems

[10] Çano, Erion. (2017). Hybrid Recommender Systems: A Systematic Literature Review. Intelligent Data Analysis. 21. 1487-1524. doi: 10.3233/IDA-163209.

[11] Y. Koren, R. Bell and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," in Computer, vol. 42, no. 8, pp. 30-37, Aug. 2009, doi: 10.1109/MC.2009.263.

[12] *Slimani, T., & Lazzez, A. (2014). Efficient Analysis of Pattern and Association Rule Mining Approaches. ArXiv. https://doi.org/10.5815/ijitcs.2014.03.9*

[13] *Zhang, S., Yao, L., Sun, A., & Tay, Y. (2017). Deep Learning based Recommender System: A Survey and New Perspectives. ArXiv. https://doi.org/10.1145/3285029*

[14] *He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. (2017). Neural Collaborative Filtering. ArXiv. https://doi.org/10.48550/arXiv.1708.05031*

[15] *Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. 2016. Wide & Deep Learning for Recommender Systems. In Proceedings of the 1st Workshop on Deep Learning for Recommender Systems (DLRS 2016). Association for Computing Machinery, New York, NY, USA, 7–10. https://doi.org/10.1145/2988450.2988454*

[16] Mishra, Nitin & Chaturvedi, Saumya & Vij, Aanchal & Tripathi, Sunita. (2021). Research Problems in Recommender systems. Journal of Physics: Conference Series. 1717. 012002. 10.1088/1742-6596/1717/1/012002.

[17] Y. Tao, R. Ma, M. -L. Shyu and S. -C. Chen, "Challenges in Energy-Efficient Deep Neural Network Training with FPGA," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 2020, pp. 1602-1611, doi: 10.1109/CVPRW50498.2020.00208.

[18] A. R. Lahitani, A. E. Permanasari and N. A. Setiawan, "Cosine similarity to determine similarity measure: Study case in online essay assessment," 2016 4th International Conference on Cyber and IT Service Management, Bandung,

Indonesia, 2016, pp. 1-6, doi: 10.1109/CITSM.2016.7577578.

[19] J. Deng, W. Dong, R. Socher, L. -J. Li, Kai Li and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 2009, pp. 248-255, doi: 10.1109/CVPR.2009.5206848.

[20] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., & Sutskever, I. (2021). Learning Transferable Visual Models From Natural Language Supervision. *ArXiv*. https://doi.org/10.48550/arXiv.2103.00020

[21] Lin, T., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L., & Dollár, P. (2014). Microsoft COCO: Common Objects in Context. *ArXiv*. https://doi.org/10.48550/arXiv.1405.0312

[22] Zeng, D., Yu, Y., & Oyama, K. (2019). Deep Triplet Neural Networks with Cluster-CCA for Audio-Visual Cross-modal Retrieval. *ArXiv*. https://doi.org/10.48550/arXiv.1908.03737