Counterfactual Explanations for Visual Recommender Systems

Neham Jain*
Carnegie Mellon University
Pittsburgh, USA

Vibhhu Sharma*
Carnegie Mellon University
Pittsburgh, USA

Gaurav Sinha
Microsoft Research
Bangalore, India

ABSTRACT

Users rely on clever recommendations for items they might like to buy, and service providers rely on clever recommender systems to ensure that their product is recommended to their target audience. Providing explanations for recommendations helps to increase transparency and the users' overall trust in the system, besides helping practitioners debug their recommendation model. Modern recommendation systems utilize multi-modal data such as reviews and images to provide recommendation. In this work, we propose CAVIAR (Counterfactual explanations for VIsual Recommender systems), a novel method to explain recommender systems that utilize visual features of items. Our explanation is counterfactual and is optimized to be simultaneously simple and effective. Given an item in the user's top-K recommended list, CAVIAR makes a minimal, yet meaningful, perturbation to the item's image-embedding such that it is no longer a part of the list. In this way, CAVIAR aims to find the visual features of the item that were the most relevant for the recommendation. In order to lend meaning to the perturbations, we leverage CLIP model to connect the perturbed image features to textual features. We frame the explanation as a natural language counterfactual by contrasting the observed visual features in the item before and after the perturbation.

CCS CONCEPTS

• **Computing methodologies** → **Artificial intelligence**.

KEYWORDS

Counterfactual Explanations, Visual Recommender systems

ACM Reference Format:
Neham Jain, Vibhhu Sharma, and Gaurav Sinha. 2024. Counterfactual Explanations for Visual Recommender Systems. In *Companion Proceedings of the ACM Web Conference 2024 (WWW '24 Companion), May 13–17, 2024, Singapore, Singapore.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3589335.3651484

1 INTRODUCTION

In the modern world, where users have no dearth of options to choose from, there is an increasing reliance on recommender systems to guide users to the best product for them. They offer a degree of personalization to the user's interactions and consequently increase user satisfaction. The use of deep learning methods, along with more and more data surrounding user-item interactions has led to the development of better recommendation models at the cost of understand-ability. Providing explanations for recommendations helps to improve the trust, persuasiveness, transparency, satisfaction, effectiveness and efficiency of recommendation systems [12, 14]. It also facilitates system designers for better system debugging. There is a growing need to ensure that users trust and understand the system, and explainability is useful in this regard.

A counterfactual explanation reveals what should have been different in an instance in order to observe a different outcome. The main advantage of counterfactual explanations over other types of explanations is understand-ability. Counterfactuals provide a causal understanding of which aspects led to which recommendations. In our problem setting, these explanations are concise, scrutable, and actionable, as they are minimal sets derived using a counterfactual setup over a user's own interactions. They also do not expose any information about other users, thus eliminating privacy concerns.

Existing counterfactual explanation models for recommender systems either do not utilize item information at all [3, 13], or only utilize information gathered from textual user reviews [11]. An item's image is often what draws a user towards it. As a result, the focus on multi-modal features in recommendation models has increased. Models like VBPR[4], DVBPR[5], DeepStyle[7], ACF[1], NPR[9] etc., all utilize item images to arrive at a recommendation. Unfortunately, current counterfactual explanation models do not work for visual recommendation systems. We make the following contributions.

- We devise a framework CAVIAR that generates counterfactual explanations for visual recommender systems. Given a user and a recommended item, CAVIAR extracts the visual features most relevant for the recommendation by finding the least perturbation to the item's image embedding that displaces it from the users recommended list.
- Our optimization algorithm intended to find the minimal perturbation described above minimizes a novel loss function that can perform a search in the image feature space and return the optimal counterfactual image features.
- We perform experiments to compare our method with baselines and demonstrate its effectiveness with respect to multiple metrics of interest. We also qualitatively demonstrate validity of our explanations by identifying parts of user reviews that align with explanations generated by CAVIAR.

2 RELATED WORK

In the domain of counterfactual explanations, PRINCE [3] defines an explanation as a set of minimal actions performed by the user that, if removed, changes the recommendation to a different item. Another recent approach ACCENT [13] uses influence functions to identify

*Both authors contributed equally to this research.

WWW '24 Companion, May 13–17, 2024, Singapore, Singapore

ACM ISBN 979-8-4007-0172-6/24/05
https://doi.org/10.1145/3589335.3651484

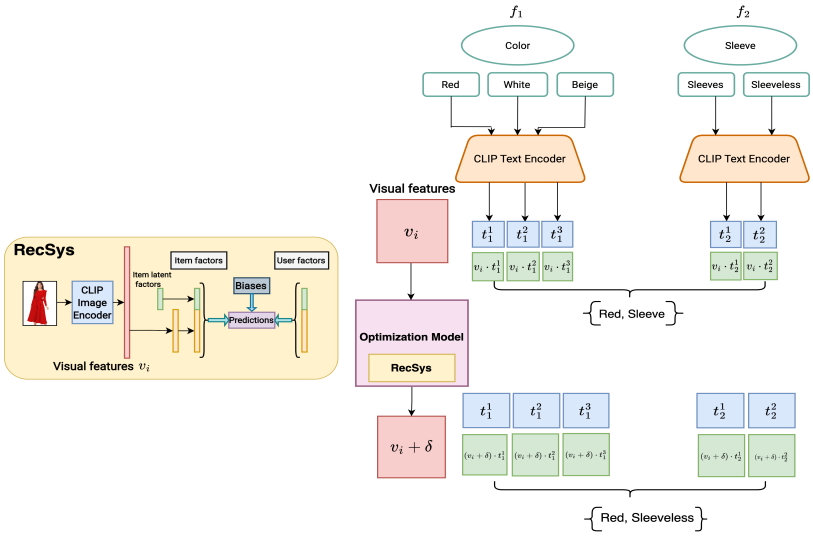

Figure 1: An overview of CAVIAR. The architecture of the underlying recommender model based on the VBPR [4] model is presented on the left hand side and that of the explanation methodology is presented on the right. The optimization model generates a minimal perturbation of visual embedding of the item's image such that the item is no longer in the top-K recommended products. A natural language explanation can be generated by comparing how the perturbed image correlates with different aspect classes as compared to the original image. For e.g., in this figure, the original image and the perturbed image both correlate with the colour aspect *Red* whereas with respect to the sleeve-length aspect, the original image correlates with *Sleeves* whereas the perturbed one correlates more with *Sleeveless*.

the training points that are the most relevant to a recommendation, from a single to a pair of items, while deducing a counterfactual set in an iterative process. Both these models are grounded in the user actions instead of item aspects, thereby making a comparison with our method unreasonable. They also do not explicitly utilize visual features. A recent counterfactual explanation method CountER [11] uses phrase level sentiment analysis to uncover item related aspects mentioned in reviews and the user's or item's score on those aspects. However, it does not make use of the information in the item's image in any way.

3 PROBLEM FORMULATION

In this section, we formulate our main problem statement. Before doing that, we introduce some notation that will be used. Consider a finite set of users U and a finite set of items V. Without loss of generality we assume $U \subset \mathbb{R}^{d_U}$ for some integer d_U. This can easily be achieved by appropriately creating features for each user either by hand or via other embeddings. For each item $v \in V$, we assume that $v = (v_i, v_o)$, where v_i is an image and v_o are other non-visual attributes of V. Similar to the users, we assume that $v_i \in \mathbb{R}^{d_I}, v_o \in \mathbb{R}^{d_O}$ for some integers d_I, d_O respectively. Next, let $\mathcal{R} : \mathbb{R}^{d_U} \times \mathbb{R}^{d_I} \times \mathbb{R}^{d_O} \to \mathbb{R}$ be a recommender model. For each user $u \in U$ and item $v \in V$, $\mathcal{R}(u, v) \in \mathbb{R}$ is a real valued score indicating how "relevant" item v is to user u. Based on this score, for each user one can rank the list of items to get a ranked list $\mathcal{L}_u$ and use the top items as a recommendation for the user.

For generation of our explanation as a natural language sentence, we assume the knowledge of a set $F = \{f_1, \ldots, f_d\}$ of image aspects. Every f_j can be further written as a set of textual classes $T_j = \{t_j^1, \ldots, t_j^{d_j}\}$. For e.g., f_j might denote the color aspect of the image and $t_j^1, \ldots, t_j^{d_j}$ might be the various possible colors e.g., *red, blue, green* etc. For this paper we will assume that v_i is obtained as an embedding of the image in v and t_k^j is a text embedding of the textual classes in the aspect f_j. Both of these are assumed to be obtained using the image and text encoder of CLIP [10] respectively. We denote the value of aspect f_j for image embedding v_i as

$$f_j(v_i) = \sigma(t_j^1 \cdot v_i, \ldots, t_j^{d_j} \cdot v_i)$$

where $p \cdot q$ denotes the dot product between two vectors p and q and $\sigma : \mathbb{R}^{d_j} \to \mathbb{R}^{d_j}$ denotes the SoftMax activation function. As a result the aspect $f_j(v_i)$ becomes a probability distribution over the classes indicating how much a certain class correlates with the image. Here is our formal problem statement.

Problem Statement: Fix an integer K. Let $U, V, \mathcal{R}$ be as given above. Let u be a user, $v = (v_i, v_o)$ be the item with highest value of $\mathcal{R}(u, v)$ and $\mathcal{L}_u$ be the ranked list of items for u as described above. We first wish to find $\delta \in \mathbb{R}^{d_I}$ with minimum ℓ_1 norm i.e. $\|\delta\|_1$ such that the relevance score $\mathcal{R}(u, v^\delta = (v_i + \delta, v_o))$ obtained on perturbing v_i to $v_i + \delta$ is smaller than the scores of the top K items in $\mathcal{L}_u$ i.e. the item falls out of the top K items once the embedding of its image is perturbed by δ. Using this δ we want to generate an explanation as a comparison of the aspect distributions $f_j(v_i)$ and $f_j(v_i + \delta)$ for every $j \in [d]$.

4 METHODOLOGY

In this section, we present our explanation framework CAVIAR. An architecture diagram highlighting all key components is provided in Fig. 1. As described earlier, our method is quite general and can explain any recommendation system that uses CLIP[10] based embeddings of item images. However, for the sake of completion, in this paper, we use a modified version of the recommendation model from VBPR [4]. The recommendation model in [4] utilizes pre-trained image features from Alex-Net [6]. We modify this part and instead use pre-trained image features from the image encoder of CLIP [10]. The architecture of this modified recommendation model can be found on the left side in Fig. 1. As described in Sec. 3, we assume knowledge of image aspects $f_1, \ldots, f_d$ and textual classes corresponding to each aspect. We use embeddings t_j^k, $j \in [d], k \in [d_j]$ of these textual classes created via CLIP's [10] text encoder (See Sec. 3).

Recall that for any user $u \in U$, and item $v = (v_i, v_o) \in V$ that is most relevant to u (i.e. has the highest $\mathcal{R}(u, v)$), we want to find vector $\delta \in \mathbb{R}^{d_I}$ with the smallest ℓ_1 norm such that $\mathcal{R}(u, (v_i + \delta, v_o))$ is not among the top K items in $\mathcal{L}_u$. Here $\mathcal{L}_u$ is a ranked list of items from V, ordered according to their relevance scores. Let v_K be the K^{th} item from the top in $\mathcal{L}_u$, i.e. it has the K^{th} largest relevance score (say r_K). In order to find a perturbation $\delta \in \mathbb{R}^{d_I}$ with the smallest ℓ_1 norm i.e. $\|\delta\|_1$, such that $(v_i + \delta, v_o)$ has a score (say $r_\delta = \mathcal{R}(u, (v_i + \delta, v_o))$) smaller than r_K (i.e. it falls out of the top K items), we can focus on minimizing the quantities $\|\delta\|_1$ and $r_\delta - r_K$ respectively. This leads us to two main components of our minimization objective, $L_1 = ReLU(r_\delta - r_K)$ and $L_2 = \|\delta\|_1$. Since our final goal is to provide an explanation by comparing the aspect distributions $f_j(v_i) = (f_j(v_i)^1, \ldots, f_j(v_i)^{d_j})$ and $f_j(v_i + \delta) = (f_j(v_i + \delta)^1, \ldots, f_j(v_i + \delta)^{d_j})$, we also minimize the overlap between the aspect distributions before and after the δ perturbation. This leads us to another component in our minimization objective $L_3 = \sum_{j=1}^{d} \sum_{k=1}^{d_j} f_j(v_i)^k \log f_j(v_i + \delta)^k$. Putting these together we arrive at our minimization objective $L = \alpha_1 L_1 + \alpha_2 L_2 + \alpha_3 L_3$, with $\alpha_1, \alpha_2, \alpha_3 \in \mathbb{R}$ being hyper-parameters that control the relative importance between the three parts. We minimize this overall objective using an iterative optimization method. Using the $\delta \in \mathbb{R}^{d_I}$ obtained as a solution to this minimization, we frame our final explanation as a natural language sentence using the textual classes that changed between the aspect distributions $f_j(v_i + \delta)$ and $f_j(v_i)$. All the aspects j, such that $argmax(f_j(v_i + \delta)) \neq argmax(f_j(v_i))$ are used to provide the explanation. For example, let the initial distribution for aspect f_1 before perturbation be $[0.92, 0.03, 0.05]$ and the distribution for aspect f_1 after perturbation be $[0.42, 0.49, 0.09]$. Since $argmax(f_1(v_i))$ is different from $argmax(f_1(v_i + \delta))$, f_1 is an aspect used to provide the explanation.

5 EXPERIMENTAL SETUP AND RESULTS

In this section, we perform experiments validating the performance and usefulness of our method.

Datasets: We evaluate our method on the publicly available Amazon Fashion Men and Women datasets [8] consisting of around 112k and 397k user-item interactions respectively. We discard all users having less than 5 interactions. Fashion is an ideal domain for testing these recommender systems because of the intricate differences between items' appearances that directly influence users.

Implementation Details: The base recommender system is trained in the same way as VBPR [4]. For our optimization model, we utilise SGD optimizer with a learning rate of 3×10^{-4}. We optimize our model for 1000 steps of stochastic gradient descent. We empirically choose the values of $\alpha_1, \alpha_2, \alpha_3$ to 1, 1 and 2 respectively. Choosing these parameters in a more data driven way is an interesting future direction. We choose a total of 5 possible aspects in our experiments which are on the basis of color, sleeve length, formal or casual, collar and fit of product. Using domain knowledge, these can be expanded further. The value of K is chosen as 10 in our experiments. We fix the maximum value of L_2 as 1. This ensures that the optimization process is stable. If L_2 is not bounded in this manner, it can cause the class loss L_3 to be overwhelmingly negative in order to compensate and this leads to sub-par optimization. We use CLIP with ViT-B/32 as the image encoder which gives an image embedding vector of 512 length.

Baselines: As a baseline for this work, we extended CountER [11] to utilize information from images. This involved creating a fixed and pre-decided set of image aspects based on domain knowledge and then scoring the image on the presence of these aspects using CLIP [10]. To provide a counterfactual explanation we perturb these features in a similarly framed optimization problem. Other details are kept same as the original CountER [11] paper. Another baseline that we compare against is random perturbation combined with our technique of creating natural language explanations. We randomly generate a vector $\delta \in \mathbb{R}^{d_I}$ with a unit norm and add it to the image feature. The rest of the method is same as ours.

Metrics: We use fidelity, explanation fidelity and explanation number to measure the effectiveness of our explanations. We define fidelity as the fraction of items recommended to the user for which the explanation model is able to remove the item from the top K recommendations. We define explanation fidelity as the fraction of items recommended to the user for which the explanation model is successfully able to generate an explanation i.e. at least one aspect differs before and after the image perturbation. A higher explanation fidelity indicates that the model is more successful at identifying relevant features and generating explanations. Explanation number is defined as the number of features required to explain a recommendation. We only calculate the explanation number for the instances where the model is successfully able to generate an explanation. A clear, concise explanation that only lists the top features responsible for a recommendation is preferred. Together, these three metrics provide a measure of the model's competency in generating relevant explanations.

Results and ablations: We report the comparative results of our experiments in Table 2 and 3. Since, modified CountER directly modifies the item aspect to provide explanation, explanation fidelity for it is always 1. Compared to other methods, our method is able to provide a more concise explanation and able to successfully generate an explanation. An increase in explanation fidelity can be obtained by increasing the number of aspects that we use to provide explanation. We perform an ablation on the loss function to show the effectiveness of using all the components. We exclude the cases

Table 1: Qualitative Analysis

Explanation by CAVIAR	Review written by the user
If the item had the color black instead of red it would not have been recommended	Got this for myself since my favorite **color is red** and I haven't been disappointed! Very professional and classy! Will buy more!
If the item had short sleeves instead of long sleeves, it would not have been recommended	Great price! These shirts are great for cold winter days but are made of cheap material. Still they are nice and comfortable long **sleeve** shirts.
If the item had no collar instead of a collar, it would not have been recommended	Husband really likes it. The long length and knit **collar** helps keeps drafts from getting inside.
If the item had the color blue instead of black it would not have been recommended	Love the fit, the **color** and the way I feel when I wear these shoes. The toning feature really makes a difference I the way I look and feel.
If the item was casual instead of formal it would not have been recommended	This pants are perfect for **formal** wear, the quality is really nice.

Table 2: Performance metrics on Amazon Men dataset

	Fidelity	Explanation Fidelity	Explanation Number
Random	0.12	0.03	3.78
Modified CountER	0.94	1	3.212
CAVIAR (L1 and L2)	1	0.04	2.78
CAVIAR (L2 and L3)	0.14	0.83	2.56
CAVIAR (L1, L2 and L3)	0.98	0.81	2.36

Table 3: Performance metrics on Amazon Women dataset

	Fidelity	Explanation Fidelity	Explanation Number
Random	0.09	0.07	2.76
Modified CountER	0.96	1	2.88
CAVIAR (L1 and L2)	0.99	0.08	2.98
CAVIAR (L2 and L3)	0.08	0.86	2.12
CAVIAR (L1, L2 and L3)	0.99	0.79	2.23

where L_2 is not present as this leads to trivial cases. Using only L_1 and L_2 is able to remove the item out of the top K recommendations but is not sufficient to generate a recommendation. Using only L_2 and L_3 is able to successfully generate an explanation but it is not able to remove the item out of the top K recommendations. Thus, using all three components L_1, L_2, L_3 yields results that are able to remove the item from the top K recommendations and also provide an explanation most of the times. We also perform qualitative evaluation by using a user's review on an item they liked as the ground truth for why they liked it. We mask the user's interaction with an item that they have previously rated well and observe the explanation given by CAVIAR when the item is recommended back to the user. We then use FLAN-T5[2] to check if the user mentioned the visual features highlighted by CAVIAR positively in his original review. The explanation provided by our model echoes the sentiment of the reviews. We provide some of these examples in Table 1. We leave the calculation of quantitative metrics based on the review to future work. CAVIAR does have some limitations: it requires a set of domain-specific visual features and can provide incomplete explanations by itself when there are essential aspects that are not visually prominent.

REFERENCES

[1] Jingyuan Chen, Hanwang Zhang, Xiangnan He, Liqiang Nie, Wei Liu, and Tat-Seng Chua. 2017. Attentive Collaborative Filtering: Multimedia Recommendation with Item- and Component-Level Attention. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Shinjuku, Tokyo, Japan) *(SIGIR '17)*. Association for Computing Machinery, New York, NY, USA, 335–344. https://doi.org/10.1145/3077136.3080797

[2] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling Instruction-Finetuned Language Models. https://doi.org/10.48550/ARXIV.2210.11416

[3] Azin Ghazimatin, Oana Balalau, Rishiraj Saha Roy, and Gerhard Weikum. 2020. *PRINCE: Provider-Side Interpretability with Counterfactual Explanations in Recommender Systems.* Association for Computing Machinery, New York, NY, USA.

[4] Ruining He and Julian McAuley. 2016. VBPR: visual bayesian personalized ranking from implicit feedback. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 30.

[5] Wang-Cheng Kang, Chen Fang, Zhaowen Wang, and Julian McAuley. 2017. Visually-Aware Fashion Recommendation and Design with Generative Image Models. *2017 IEEE International Conference on Data Mining (ICDM)* (2017).

[6] 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* (Lake Tahoe, Nevada) *(NIPS'12)*. Curran Associates Inc., Red Hook, NY, USA, 1097–1105.

[7] Qiang Liu, Shu Wu, and Liang Wang. 2017. DeepStyle: Learning User Preferences for Visual Recommendation. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Shinjuku, Tokyo, Japan) *(SIGIR '17)*. Association for Computing Machinery, New York, NY, USA, 841–844. https://doi.org/10.1145/3077136.3080658

[8] Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying Recommendations using Distantly-Labeled Reviews and Fine-Grained Aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, 188–197. https://doi.org/10.18653/v1/D19-1018

[9] Wei Niu, James Caverlee, and Haokai Lu. 2018. Neural Personalized Ranking for Image Recommendation. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining* (Marina Del Rey, CA, USA) *(WSDM '18)*. Association for Computing Machinery, New York, NY, USA, 423–431. https://doi.org/10.1145/3159652.3159728

[10] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 8748–8763.

[11] Juntao Tan, Shuyuan Xu, Yingqiang Ge, Yunqi Li, Xu Chen, and Yongfeng Zhang. 2021. Counterfactual explainable recommendation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*.

[12] Nava Tintarev and Judith Masthoff. 2007. A Survey of Explanations in Recommender Systems. In *2007 IEEE 23rd International Conference on Data Engineering Workshop*.

[13] Khanh Hiep Tran, Azin Ghazimatin, and Rishiraj Saha Roy. 2021. Counterfactual Explanations for Neural Recommenders. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

[14] Yongfeng Zhang and Xu Chen. 2020. Explainable Recommendation: A Survey and New Perspectives. *Found. Trends Inf. Retr.* 14, 1 (mar 2020), 1–101. https://doi.org/10.1561/1500000066