Novelty in Content Creation: Experimental Results Using Image Recognition on a Large Social Network

Abstract

Social networks rely on sharing engaging content with their users. Since continued production of user-generated content is critical to their success, they have constructed a variety of tools to motivate new content creation, to facilitate user discovery of new content, and to provide attention and recognition to the best user-generated content. Past research shows that such attention and recognition increase the volume of content shared on the networks. But how do these affect the *nature* of content shared on their platforms? Do they cause creators to share content similar to the ones that received attention and recognition? Or do creators take risks and create different content than the ones recognized? These are the questions we ask in this paper. Our empirical context is an image-sharing social network where creators share digital art and photography. We leverage a randomized controlled experiment to induce exogenous variation in attention and recognition to specific content. Using a transfer learning-based machine learning algorithm, we convert complex images into lower-level features. This allows us to analyze similarities and differences between images. Our main findings are that creators produce and share different content on the social network than the ones that received attention and recognition. This result is robust to a variety of ways in which we classify image content. Our results illustrate the importance of tools aimed to induce attention and recognition to the creation and development of diverse content by social media creators and give insights into factors that motivate content creators to create content.

Keywords: User-generated content, machine learning, transfer learning, image recognition, field experiments, award recognition, attention

1 Introduction

Social networks rely on user-generated content to attract and retain users on their platforms. These networks increasingly face stiff competition from other platforms for users' time and attention and are therefore motivated to cultivate novel and unique content on their platforms. Traditional media can more directly influence content novelty since the content is either produced by the media outlets themselves or contracted out to third-party content producers. By contrast, social media has to rely on indirect levers, for example, front pages, spotlights, featured content, news feeds, and trending tabs, to direct attention and recognition to the best pieces of content and to motivate their users to create more content and create more novel content. Past literature has documented that attention and recognition lead to more engagement and greater content creation by users whose work received that attention and recognition on social networks (Toubia & Stephen 2013; Muchnik *et al.* 2013; Huang & Narayanan 2021) and content sites like Wikipedia (Zhang & F. Zhu 2011; Aaltonen & Seiler 2015; Kummer 2013; Gallus 2017; K. Zhu *et al.* 2020).

However, the impact of these levers on the *nature* of the work created by users is less clear. On the one hand, attention and recognition can act as external signals to the content creator of the quality and popularity of their work, and can motivate them to create more content similar to the one that got them attention and recognition. On the other hand, these can satisfy their need for external validation from their audience and license them to explore their interests, take more risks and create content different from the ones that got them attention. In this study, we examine the impact of attention and recognition generated through the featuring of content on the front page of a social network on the nature of the work that users create subsequently. We leverage a field experiment run in collaboration with an image-sharing social network, where attention and recognition are exogenously manipulated for a randomly selected set of users to find causal evidence to answer our main research questions.

Content creation has been considered to be a function of both extrinsic and intrinsic motivators (Toubia & Stephen 2013; Muchnik *et al.* 2013). Lerner & Tirole (2002) refer to two incentives for creators to create and share open source software content - the *career enhancement incentive* and the *ego gratification incentive*. The former refers to the ability of the content creator to signal their quality and capabilities in the specific domains in which they create content, while the latter refers

to the desire on the part of content creators to generate recognition from their peers. These are both examples of extrinsic motivators for content creation. The behavioral literature has also explored the effects of extrinsic motivators such as peer recognition and attention, through their impact on the affective state of the creator, and thereby their desire to continue their positive affective state through creation of more content (Isen *et al.* 1987). Indeed such extrinsic motivators have been empirically identified as key drivers of content creation in a variety of contexts such as open source software (C.-G. Wu *et al.* 2007), online communities and forums (Jin *et al.* 2015) and sales-force motivation (Larkin 2011; Ederer & Manso 2013).

Content creators also have intrinsic motivators to create content. The process of creating content might be inherently pleasurable for the creator. And they might derive utility from sharing what they produce since they might consider it as a public good - something others could use and build on in the context of open-source software, for instance.

Given that creators have both extrinsic and intrinsic motivators for content creation, the prediction of the effect of recognition and attention on the nature of future content created and shared by the users is ambiguous. On the one hand, extrinsically motivated creators may see attention and recognition from their peers as signals of quality of the work. Since they would likely wish to receive more attention and recognition, these signals should motivate them to subsequently create and share work similar in nature to the one that received attention. On the other hand, if they perceive the external audience as desiring variety in content, or if they see that their perceived quality among their peers depends on demonstrating their versatility in creating different kinds of content, they would be motivated to create works different from the ones that generated attention.

Purely intrinsically motivated creators would typically not be impacted by external recognition or attention in the nature of content they produce. However, most creators are likely to have a mix of intrinsic and extrinsic motivations for creating and sharing content. Motivational crowding theory suggests that the introduction or elevation of extrinsic motivators could decrease the effect of intrinsic motivators (Frey & Jegen 2001; Gneezy *et al.* 2011), suggesting that recognized creators might be discouraged from producing content they find intrinsically interesting. Alternatively, if the effects of extrinsic motivators such as attention and recognition are concave in nature, creators whose extrinsic motivators have been satiated should focus on creating content that gives them intrinsic utility. This could lead to content creation different from the ones created for satisfying the extrinsic

part of utility, and which in turn generated attention and recognition. Thus, consumers with intrinsic motivation (but also some degree of extrinsic motivation) would create content different from the one that generated attention and recognition if these satiate their extrinsic motivators.

A related empirical work on award recognition's impact on content novelty is Burtch *et al.* (2022), which experimentally gifted Reddit Gold awards to users on the platform and found that the provision of these awards decreased treated users' subsequent content novelty. However, we highlight a few notable differences between their study and the present work which could lead to divergent conclusions on the effects of award recognition on the content novelty of treated users. First, their context is a text-based discussion platform whereas this research focuses on image-based art platform. In addition to potential differences arising from the type of content itself, there are potentially higher expectations around what constitutes novelty in an artistic context and heightened barriers to content creation. Secondly, their study awarded relatively average users selected among almost all posters in their focal communities, while in contrast our experimental users were selected amongst the best creators on the platform. Burtch *et al.* (2022) found that while content novelty of treated users decreased, their intervention did ultimately stimulate the creation of novel content through the mechanism that newer users with comparatively novel content were stimulated to create more content. Our context focuses on seasoned, top creators who are more likely to have received previous forms of external recognition. Therefore, relative to the users in that study, the users in ours are less likely to be influenced in their beliefs about what kind of work they are good at through this external recognition, and hence less likely to create content similar to the recognized work. Finally, there are significant differences in the nature and prestige of the awards that have the potential to influence how it is perceived by the recipient. Burtch *et al.* (2022) utilize a peer award which is common, largely cosmetic in nature, and has minimal impact on the visibility of the awarded content. In contrast, our study's award is the highest recognition bestowed by the platform, pushes notifications to all the creators' peers, and elevates the work to front page visibility in front of thousands of viewers. These differences are large enough that, while both nominally considered awards, they represent distinct mechanisms by which platforms direct attention to content. For the reasons described, it is unclear how the prior work's findings should translate into our context.

Another related study of the impact of recognition on content novelty is Negro *et al.* (2022),

which examines the impact of a status-conferring award - the Grammy Award - on subsequent content creation by musicians. The main point of that paper is that artists who received the Grammy Awards created music that was more different on various pre-defined characteristics to those created by contemporaneous artists than artists who were award finalists but did not win. There are a few aspects of this study that we would like to highlight that distinguish our work. First, the analysis of the impact of the Grammy Awards is based on non-experimental, observational data. The challenge with this is that the causality of the documented effects is hard to establish. The study attempts to mitigate this concern in robustness checks using a matching estimator but cannot eliminate concerns about the validity of the necessary assumptions for this approach to work (**??**). The "control" group consists of artists who were finalists for the award in a given category, but did not win the award. It is not possible to distinguish their subsequent actions as a consequence of not winning the award from the actions of the "treatment" group, i.e. award winners. Thus, it is difficult to know if award winners created more novel content in response to winning the award, or the losers created less novel content in response to not winning the award. By contrast, our study uses a controlled field experiment to exogenously affect attention and recognition to the treatment users' content. Thus, we can provide conclusive causal evidence of the effects we aim to find. Second, the content creators in the control group of our experiment do not know that they are in the control group - this mitigates concerns about SUTVA violations. We additionally conduct analysis to support the idea that our results are not affected by this issue.

In summary, it is unclear how the attention and recognition as a result of platform levers affects the nature of content produced by creators - in particular, it is unclear if these will cause the creator to create content similar or dissimilar to the ones that generated attention and recognition. Social media platforms have strong preferences for content variety due to the demands of user engagement (F. Wu & Huberman 2007; Huotari & Ritala 2021; Ciampaglia *et al.* 2015). Ideally for these platforms, the visibility mechanisms (such as front pages) that drive user engagement to top creators would also encourage these top creators to continue creating novel content.

Content novelty is subjective and difficult to measure in user generated content, which often comes in the form of unstructured expressions such as images or video. Human judgments of images are subjective, and the typical social media context displays images alongside peer feedback, leading to herding behavior (Muchnik *et al.* 2013) and unreliability of aggregate signals such as

like counts. Finally, the assignment of recognition and visibility is typically endogenous on social networks. To the extent that a social network utilizes human judgment in determining what content to feature, the curated content will tend to differ systematically from content that is not chosen for featuring. Even the absence of manual curation, algorithms for content discovery utilize user signals such as views, likes, or engagement to identify content to promote. The non-random selection of content makes establishing counterfactual user behavior in the absence of recognition difficult.

This paper addresses these challenges through the implementation of a randomized field experiment on a large, artistic image-sharing social network. This social network, *Behance*, has over 5 million users who have created over 9 million digital albums of images called 'projects'. This platform is host to highly artistic and creative professionals, including digital artists, illustrators, photographers, animators, and more. *Behance* utilizes human curators to identify exemplary content to feature on the front page of the website, where the featured content can be seen by all visitors to the site and is awarded a badge of recognition. Through collaboration with the site, we introduce randomization to which content ultimately gets featured from among a larger set of curated candidate content pieces, allowing for clean comparison between treated users, whose content is featured and control users, whose content is not.

We look to characterize how feature recognition and the resulting peer attention and recognition directed by it can impact the nature of the subsequent creative output of a content creator. To do so, we begin by downloading the images created by users in the experiment. These images are pieces of digital art, and are highly complex in nature. Similarity and dissimilarity between pairs of images is thus challenging to measure, especially with existing image recognition tools that have been trained on photorealistic objects. Added to this is the fact that we have thousands of images in our dataset, which renders a fully manual process of assessing similarity impractical. We instead look to a transfer learning approach that can extend a small set of human labels specialized to our creative context of digital art while leveraging the deep learning from a much larger set of images.

We collect training data for transfer learning by soliciting labels and ratings on a smaller, random subset of images which were human-labeled via Amazon Mechanical Turk (MTurk). Workers label the images on seven pre-defined dimensions and provide open-ended keywords describing the image. The base for our transfer learning is Inception Net (Szegedy, Liu, *et al.* 2015), a widely popular multi-class image classification algorithm for image recognition or detection, employing deep learning

to learn about image features.

The key contribution of this model is that it achieves high levels of prediction accuracy with a much smaller dataset and lower number of parameters (still about 25 million). If we were to train an Inception Net model from scratch, we would need a very huge set of labeled training images for the model to learn all these parameters from the data. This would be very expensive, time-consuming and would defeat our very purpose of building a machine learning prediction model in the first place. Therefore, we employ the transfer learning approach where we use the weights (parameter values) from an Inception v3 model (Szegedy, Vanhoucke, *et al.* 2016)) pre-trained on the ImageNet database (Fei-Fei *et al.* 2009). Because ImageNet is a vast and diverse dataset of images (more than 14 million images), the weights learned by a deep learning model trained on ImageNet should be generic enough to work for our target images.

We make two sets of predictions for our images, first we predict the score (rated on a discrete scale of 1 to 7) of each image on seven pre-defined dimensions (abstractness, commercial intent, creativity, complexity, emotiveness, likeability, photorealism). In order to predict the image score across these predefined dimensions, we use a deep learning model coupled with these pre-learnt Inception Net parameters via transfer learning. We build one deep learning model each for the seven dimensions, and each model predicts the dimension score of the image on a discrete cardinal scale of 1-7. We modify the Inception Net model to incorporate the cardinal nature of our labels, which is different than the categorical data for which the model was originally built. Second, we also predict the keywords associated with the image. In order to predict the image keywords, we again use the same transfer learning Inception Net model but in order to deal with the very high dimensionality of the keywords space, we also use the Word2Vec model pre-trained on the Google News data. Word2Vec is a popular technique for natural language processing which uses a neural network to learn word associations from a large corpus of text. The model achieves this by creating word embeddings, which basically are words represented as lower dimensional numeric vectors in such a way that similar words have a similar vector representation. The Word2Vec model helps us represent the thousands of human-labeled keywords for our training data into a much more compact representation of a 300 dimensional vector. This lower dimensional representation of the keyword labels, coupled with the pre-trained Inception Net weights enables us to make predictions of the keyword labels for all the images in our experiment, even with a small training data size. Finally,

in addition to generating the keyword predictions for all the images using our model, we also use Google's Vision API to predict the keywords for our images. Vision API uses pre-trained deep learning models for image recognition and generating labels for the images. This provides us with an additional set of keyword labels for all the images in our experiment.

We find that users whose works are featured create significantly different and more novel content than those whose content were not featured. These findings are robust to the machine learning methods we utilize to assess image similarity, including object labeling through the Google image recognition API, transfer learning of seven human-labeled predefined image dimensions, and transfer learning of human-labeled keywords. These results represent encouraging findings for social networks hoping to cultivate novel content through the usage of attention and recognition levers, as they suggest that even highly creative individuals can benefit from these interventions.

2 Empirical Context and Experimental Design

To study the effect of attention and recognition on subsequent content creation activity, we partner with the large art-sharing social network *Behance*, which is part of Adobe Inc. Users on *Behance* create and share their artistic work with each other in the form of albums called "projects" that contain collections of images. *Behance* is a uni-directional social network similar to *Twitter*, where users form one-sided following links. Followers of a user get notified for various events related to that user, including the posting of new content, commenting activity, when one of their works has been featured, etc. Figure 1 is a screen shot of the front page of the social network. Each work shown on the front page has been selected to be featured on the site. As can be seen in this figure, the network displays collections of images for users to view and get inspired by, and for creators to showcase their work and receive likes, feedback and comments from other users.

The network provides a variety of tools for users to discover content. An important tool that is used by *Behance* to showcase work and promote discovery of high quality content is that it employs a team of human curators to browse content on the website, and select works to be 'featured' on the front page. The message at the bottom right of Figure 1 refers to this as well, pointing to the 'hand-picked' nature of the content shown on the network. This aspect of the network is similar to tools employed by other networks, such as trending content on YouTube and featured content

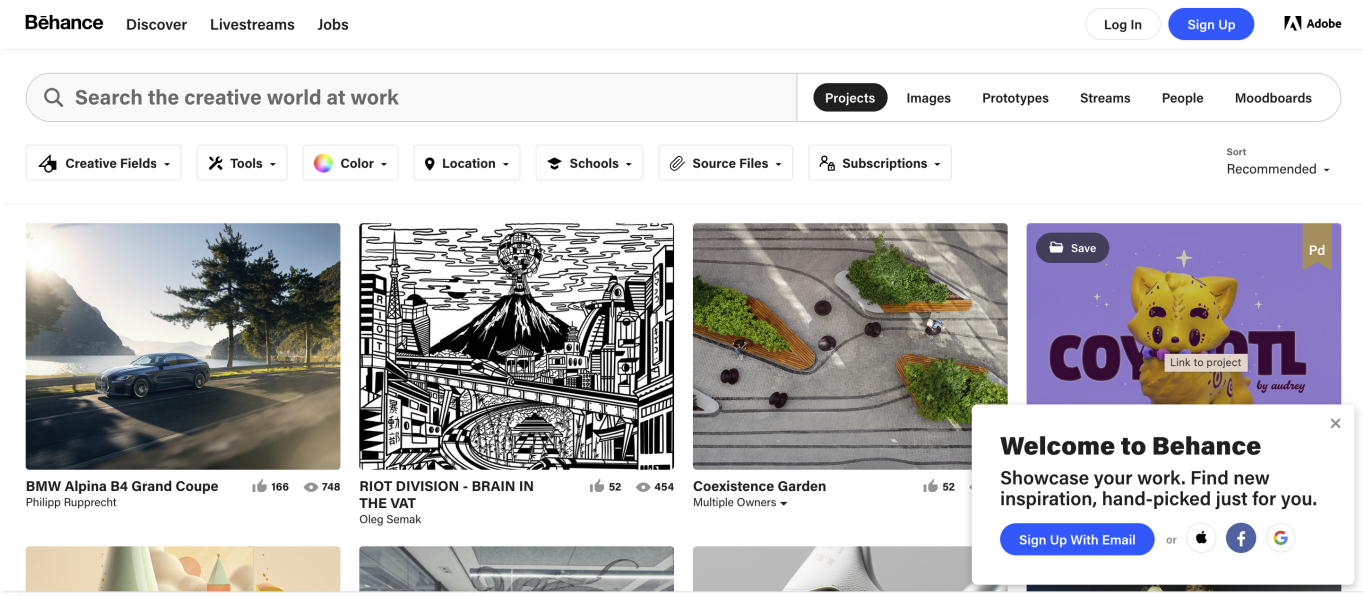


Figure 1: Screenshot of the *Behance* Front Page

on Twitter. On *Behance*, content is selected by the curators to be featured on the front page of the network every day. Featured content receives a permanent star mark, and followers of the user whose work got featured get notified about it. The permanent mark distinguishes the content as exemplary and is seen by users on the network and managers at *Behance* to be one of the highest forms of recognition available on the network. It draws attention from both followers of the user who receive news notifications of the feature and a vast number of non-followers who see the work courtesy of its placement on the front page of the website.

Figure 2 shows an example of featured and non-featured content from a user on the network. The image on the left was featured on the network, as seen by the star badge at the bottom of the image. The image on the right was not featured, and does not have this star badge.

Our experiment introduces randomization to which projects receive feature awards during the experiment and subsequent observation period. As part of the operation of the website, *Behance*'s curation team regularly combs through user generated projects to identify exemplary artwork worthy of receiving a feature award. Identified works are placed in the feature queue, an ordered list of projects scheduled to be featured on the site. At set intervals throughout the day, the project at the front of the queue is given its feature award, then removed from the queue. The randomization procedure proceeds as follows: from the (at the time of randomization) 8,921 in the feature queue, N

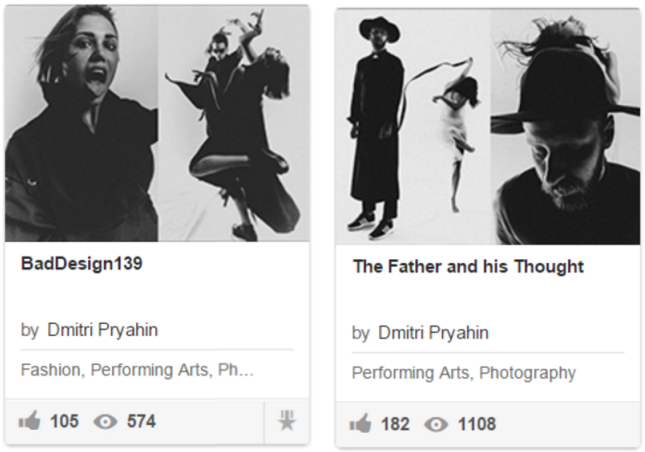
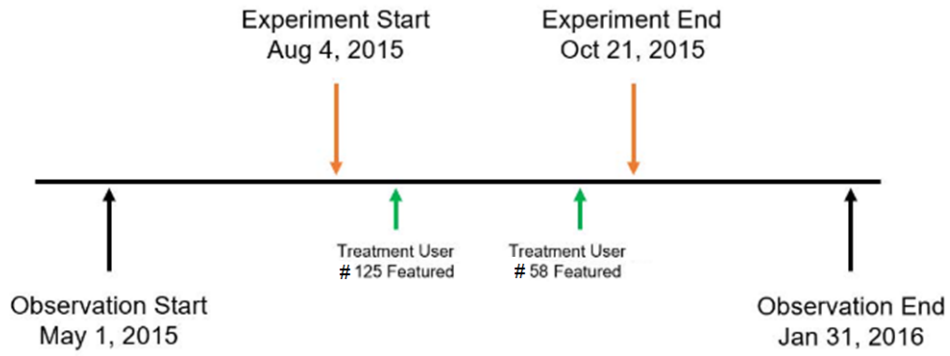


Figure 2: Example of Featured (Left) and Non-Featured (Right) Content on *Behance*

= 658 projects owned by unique users were selected for the experiment. These projects were divided into equally sized treatment and control groups of N = 329 each, and the feature queue was then rearranged such that the treatment group projects would be awarded on a random day over a two and a half month experimental period while the control group projects were not awarded during the experimental period or subsequent three month observation period (they received their award after the observation period). Other projects owned by users in the experiment were not affected by this randomization, thus the treatment can be thought of as adding one additional award to the treated users. A diagram of the experimental timeline can be seen in Figure 3. We conduct randomization checks and find no significant pre-treatment differences between treatment and control groups in either recorded network activity or labeled characteristics in Table 1.

Figure 3: Experimental Timeline



Notes: Diagram depicting the timeline of the experiment, including pre-experimental observation period, experimental period, and post-experiment observation period. Treatment group users (for example, #58 and #125 as shown) received a feature award on their focal project at a random date during the experimental period. Control group users were guaranteed to not receive an award on their focal project until the end of the post-experiment observation period.

Table 1: Randomization Checks on pre-treatment variables. Network activity quantities represent day-level measures averaged across pre-treatment days by user and then by experimental group.

	Mean of Control	Mean of Treatment	p: Two-sided t-test
Labeled Characteristics			
Abstract	3.701	3.736	0.338
Commercial	3.929	3.770	0.809
Complex	3.825	3.935	0.759
Creative	4.310	4.417	0.693
Emotive	3.813	3.897	0.561
Likeability	4.433	4.528	0.680
Photorealism	4.215	4.383	0.754
Cosine similarity to focal project	0.449	0.426	0.126
Euclidean distance to focal project	2.452	2.372	0.477
Dot product with focal project	3.856	3.296	0.556
Other Network Activity Measures			
Appreciations received	10.233	11.203	0.513
Comments received	0.600	0.794	0.119
Views received	111.626	119.486	0.655
Inbound ties	4.163	4.890	0.241
Appreciations given	0.972	1.064	0.783
Comments given	0.237	0.303	0.516
Views given	5.707	5.976	0.757
Outbound ties	0.194	0.240	0.299
Projects published	0.018	0.024	0.123

Note: *p<0.1; **p<0.05; ***p<0.01

Thus, we induced random variation in featuring, allowing us to the study the impact of the feature award on subsequent user behavior. Huang & Narayanan (2021) study the impact of featuring on the engagement of the users, their subsequent content creation and sharing activity on the network. The main finding of that paper is that users whose work was featured increased their engagement on the network, shared more content, and created more content after the feature award. In this paper, we leverage this same experiment to study the impact of feature awards on the nature of content created by the users. In particular, we study whether users create and share content that is similar to the ones that were featured, or whether they create and share more novel content.

3 Empirical Strategy

As noted earlier, we aim to study the impact of featuring, and the recognition and attention generated by it, on users' subsequent content creation process, and in particular whether they create and share more novel content. For this purpose, we need to be able to measure the novelty of future content. This is a challenging endeavor in our context of digital art. Consider the three images in Figure 4. These three images were pieces of digital art created by the same artist. If we were to consider the image in the middle and compare it to the images on the two sides, it is challenging to decide which of the two images - on the far left or the far right - is closer to the one in the middle. On the one hand, the image on the far left has a realistic face like the one in the middle, while the one on the right does not. On the other hand, the image on the right has flowers, like the one in the middle and unlike the one on the right. This set of images illustrates the complexity of measuring novelty of a piece of content relative to the featured content. It is not possible to measure novelty using either a direct comparison of images, or even an identification of objects in the image.

Our approach to this problem is to two-pronged. First, we convert the image into a bag of words describing it. This allows us to compare content based on similarity between these collections of words. This takes us away from literal characteristics of the images towards the meaning conveyed by the images. The second approach is to identify important pre-defined dimensions that differentiate images on this social network, obtain measures for the images on these dimensions and then measure similarity or dissimilarity between pairs of images on these dimensions.

Before we describe how we go about these two different approaches, we describe the data that

Figure 4: Content Novelty

we obtain for the purpose of our study.

3.1 Image Dataset

We downloaded a total of 37,927 images owned and created by the 658 users in the experiment over an observation period containing three months pre-experiment, two and a half months during the experiment, and three months post-experiment. These images were downloaded alongside their project format and characteristics. As described earlier, content is organized on *Behance* in projects, with each project consisting of a collection of images. The dataset comprises of 658 projects, each owned by a user in the experiment. They would be featured or not during the experimental period depending on the user's group assignment. These users created an additional 2763 projects during the observation period. Our approach is to compare the experimental images to the non-experimental images in terms of image similarity measures, and assess the causal effect of featuring on similarity for projects that were released subsequent to the project owner having their project featured as a result of the experimental manipulation. In other words, we look at projects created after the experimental project for each user, and compare the similarity of post-experimental project images to those in the experimental project for the treatment vs. control group users. In this analysis, we allow for correlations in content within users and at given points of time across users through the inclusion of appropriate fixed effects in the respective regressions, and also cluster our

standard errors appropriately to allow for correlated unobservables within projects. The fact that the variation in featuring is experimentally induced allows us to make causal inferences about the effect of featuring on similarity or dissimilarity of future content created.

For the first prong of our empirical strategy, we need to convert each image into a bag of words. We first employed a pre-trained algorithm to do this. Google's Cloud Vision algorithm provides just such an algorithm. This is an image tagging algorithm trained on a large tagged dataset of images called 'ImageNet' (Fei-Fei *et al.* 2009). The output of this algorithm is a large collection of words for every image. The algorithm is primarily trained to identify objects in images. Google provides an API to access this algorithm, taking as an input the image files, and outputting the set of words associated with each image.

As described earlier, a comparison of objects present in two images might provide an imperfect measure of dissimilarity or similarity between two pieces of digital art. Since the Google Cloud Vision algorithm is trained to primarily identify photorealistic objects, this may provide a sub-optimal measure for our purposes of labeling and characterizing creative digital artwork. This is because the ImageNet database of images is a collection of photographs of objects, including living things and inanimate objects. In contrast, our dataset has a range of vector artworks, illustrations, and fantastical imagery. And when it does contain photography, the subject matter is often depicted in an artistic manner to create an emotional impact on the viewer beyond the presence of a particular subject. Therefore, we create our own training dataset, employing the Amazon MTurk platform to employ a set of humans to label a sub-sample of our database to include not just objects or subject matter, but also actions, emotions, and artistic techniques. It would be prohibitively expensive for this project for all images in our dataset to be manually tagged. Hence, we do this for a sub-sample to create a training dataset. We then employ a transfer learning approach, starting with a pre-trained deep learning algorithm to identify basic features of images, and adding on components that are trained on our training dataset to label all our images.

The second prong of our empirical strategy involves identifying key features that we a priori consider to be discriminating characteristics of images on the *Behance* social network. These were identified by us in collaboration with the managers at the social network, as well as through a reading of the literature on characterizing art adapted to the context of digital illustrations and photography (Tinio & Gartus 2018; Goude & Derefeldt 1981; Zujovic *et al.* 2009). We identify 7 dimensions

that characterize the stylization of the work (Abstractness, Commercial Intent, Complexity, and Photorealism) and its emotional impact on the viewer (Creativity, Emotiveness, and Likeability) in terms that were both readily accessible to lay participants and tailored to *Behance*'s context which centers around digital photography and illustration by professional artists. We also utilize MTurk workers for this task, following a process similar to the one described above for labeling the images. Collecting data on these 7 dimensions for all images would be prohibitively expensive. Therefore, we used the same MTurk sample of human subjects to rate images in the training sample described earlier on these dimensions. We then use a transfer learning algorithm once again to predict the scores on these 7 dimensions for every image in our dataset.

3.2 MTurk Survey Details

We utilize Amazon MTurk to aid in manually labeling a randomly selected subset of 882 images created by users in the experiment. Each image was rated by 5 independent workers, with restrictions placed such that no worker could label more than 100 images. A total of 333 MTurkers were recruited for the task, providing a total of 4,410 image labels. Participants were given an overview of the task, asked to review an example image, then shown example keyword labels for the example image. Once they confirmed their understanding of the task, they viewed the focal image and provided 8 or more terms or keywords that could be used to categorize the image, which could include objects in the image, actions that the subjects in the image are performing, techniques describing the image or its creation, or emotions that the image depicts or is designed to elicit from the viewer. Finally, participants rated the image on each of the 7 dimensions of Abstractness, Commercial Intent, Complexity, Creativity, Emotiveness, Likeability, and Photorealism on a scale from 1-7. These terms were defined and described for the workers to aid in their ratings. A screenshot of the survey instructions is shown in Figure 5.

3.3 Details of the Transfer Learning Approach

Great efforts have been put lately into creating computer-based image processing solutions for facilitate a better understanding of artistic images (Stork 2009, Cornelis *et al.* 2011). One of the easy methods to find the similarity or differences in the pictures is by using off-the-shelf APIs such as Google's Cloud Vision. This pre-trained computer vision software labels images by detecting

Figure 5: MTurk Survey

You will be asked to provide 8 or more keyword labels for the displayed image. Please provide one keyword per text box. You may think of these keywords as labels that might be provided to categorize, tag, or index the image. These keywords can describe objects in the image, actions that the subjects in the image are performing, techniques describing the image or its creation, and emotions that the image depicts or is designed to elicit from the viewer. Please be as complete as possible.

Please review these examples to help understand the task. When you have finished reviewing the examples, click "I have read the examples and understand the task – Continue".

Example Images and Labels:

Notes: MTurk Survey provided to participants for human labeling of 882 images. Participants were shown an image created by a user in the experiment and asked to rate the image on a 1-7 scale for 7 defined dimensions, including abstractness, commercial content, complexity, emotiveness, likeability, and photorealism. They were then asked to provide 8 or more keyword tags for the image related to objects in the image, subjects in the image, techniques utilized in its creation, and emotions depicted or intended to be elicited in the image.

keywords that are associated with the image. An example[1] of the labels that these models provide can be seen in Figure A1. In fact, we use this method of auto-labelling our images.

However, it is important to note that Google Cloud Vision and other similar algorithms are trained on generic image databases that are not geared specifically for labeling creative images. The supervised learning methods that Google uses can train a model to recognize the patterns and content in images, but the images that we are dealing with on the *Behance* platform are by design creative and artistic in nature. Hence there might be salient attributes of images that are relevant to the *Behance* platform, but these canned algorithms might not be able to pick those up. For example, brand details in the images might be relevant to signify commercial intent of the image or allude towards the image being a commissioned artwork. Similarly, some of the artistic pieces might be very abstract when compared to the usual images which the auto-labelling algorithms are trained on. Hence to ensure that the key attributes of the Behance image data are retained in the labelling exercise, we build an image labelling model that is trained on our own data. This ensures that the model is geared towards detecting patterns in artistic data like the ones shared on Behance.

In order to develop our own labelling model, we would need to train a CNN on a manually labelled random sub-set of our creative image data. However, to train a full-scale deep learning model with millions of parameters from scratch, would require a huge pre-labelled training data-set of images. This will defeat the very purpose of developing the model for us because the whole point of creating this CNN model is to automate the labelling task for our data-set. Hence to circumvent this problem, we employ a transfer learning approach. Transfer learning enables us in taking the knowledge learned from CNNs that are already trained on bigger data-sets and transferring that knowledge to a model with a smaller training data-set. The idea behind this technique is that the initial layers of the CNN detect low-level features of the images such as edges and gradients, which are more transferable between different types of images. On the other hand, the later layers of the network detect features that are more specific to the creative and artistic images in our dataset and these features can be learnt based on the training data-set (Yosinski *et al.* 2014).

Transfer learning has been used in the existing literature for predicting artistic images. Lecoutre *et al.* (2017) use a dataset of paintings to train a deep learning model on detecting artistic styles. Though they have a labelled dataset of 80,000 images, it is not sufficient to train from scratch a

[1]Source - Google AutoML Vision Guide https://cloud.google.com/vision/automl/docs/beginners-guide

model that can provide high accuracy of prediction. Their model uses weights pre-trained for object recognition on ImageNet. Tan *et al.* (2016) use a similar transfer learning approach for classification of fine-art paintings. The weights for their model are pre-trained on ImageNet and the last softmax layer is retrained based on their artistic data for style recognition. We follow a similar technique to Tan *et al.* (2016) in our algorithm.

In our case, in order to label the images, we employ transfer learning on the Inception Net v3 model. For the initial layers of the Inception network we use the same weights as a model trained on a bigger data-set (ImageNet, which contains more than 14 million images) and freeze those weights. Next, we make the weights of only the last layer variable and train them based on our self-constructed data-set of Behance images. This method is suitable for us because it reduces the number of weights that need to be learnt from our training dataset that contains only a few hundred images.

We employ this deep learning model for predicting the labels for the entire set of images posted by the users in our experiment. Our relatively small total set of 875 images labelled by the MTurkers is used to train our prediction model (we use a random sub-sample of 825 images to train the deep learning models and set aside 50 images as the test set). Once we train our deep learning models on the labelled data-set, we then deploy these models to predict the labels for the unlabelled set of images in our experiment.

There are two sets of labels that need to be predicted for the images - first, we label the images on 7 different dimensions (abstractness, commercial intent, complexity, emotiveness, likeability, photorealism), and second, we predict the keyword labels for the images. Next, we describe both these models in detail.

3.3.1 Dimension Predictions

Our first prediction task is to detect the score of each image across the 7 dimensions - abstractness, commercial intent, complexity, emotiveness, likeability and photorealism. The score is a specific rating give to each image against each of the above specified dimensions, as measured on a discrete cardinal scale ranging from 1 to 7. We use an Inception Net based model (Szegedy, Liu, *et al.* 2015) to do these predictions. Inception v3 [2] is an image classification algorithm for categorical data.

[2] https://cloud.google.com/tpu/docs/inception-v3-advanced

It is a widely popular, state of the art algorithm used for classification of complex data. The key contribution of this model is that it achieves high levels of prediction accuracy with a relatively small number of parameters.

If we were to train an Inception Net model from scratch, we would need a very huge size of labelled training images for the model to learn all these parameters from the data. This would be very expensive, time-consuming and would defeat our very purpose of building a machine learning prediction model in the first place. Hence we employ a transfer learning approach where we use the weights (parameter values) from an Inception v3 model (Szegedy, Vanhoucke, *et al.* 2016) pre-trained on the ImageNet database. The ImageNet[3] database is a very large visual database which consists of more than 14 million images labelled across 21,841 categories (Fei-Fei *et al.* 2009). These categories in the ImageNet database might not be directly relevant to our requirement.

However, because ImageNet is a vast and diverse enough data-set of images, the weights learned by a deep learning model trained on ImageNet should be generic enough to identify basic features for our target images (Yosinski *et al.* 2014). Hence, in our model we use a simple algorithm with pre-learned weights for all layers (using Inception Net trained on ImageNet) in the CNN except for the last one which is a fully connected layer with a standard softmax function to output the final probabilities across the 7 ratings. The parameters corresponding to only this last layer are learnt using our training dataset.

We train 7 separate deep learning models, one each for the 7 dimensions. Since the value of each dimension varies on a discrete cardinal scale of 1 to 7, the objective of each prediction model is to predict the score on a 1 to 7 scale for a specific dimension. For example - the model predicting the "Abstractness" dimension provides a prediction of the Abstractness score for the image on a 1 to 7 scale. Hence our problem is a multi class classification problem.

We use the standard cross-entropy loss function as below,

$$L(\hat{y}, y) = -\sum_{j=1}^{7} y_j log\hat{y_j} \tag{1}$$

Where y is the input label vector of the image, $\hat{y}$ is the predicted label vector for the image. Our model's objective is to maximize the above defined loss function, so that it can minimize

[3]https://image-net.org/

the distance between the MTurker labelled score (real value) and the model output for the score (predicted value).

The architecture for each of our 7 deep learning models is the same and is described in Table 2 (it uses the Inception v3 architecture by Szegedy, Vanhoucke, *et al.* 2016). Note that the feature weights of the portion of the CNN architecture lying between the 2 red lines in Table 2 are obtained using the transfer learning approach from the Inception v3 model pre-trained on the Image Net database [4]. These weights are applied to the input image to generate a 2048 dimensional output vector. This vector is supplied as input to the final Softmax layer in Table 2 that is then trained using our given database of images to predict the final result. The final output of the Softmax layer is a vector that provides the likelihood (probability) associated with each of the 7 scores (Rating from 1 to 7) for the input image. For the purpose of predicting the final score for each image, we take a weighted average of the 1-7 discrete ratings with their probabilities as predicted by our model.

Table 2: Inception Net with Transfer Learning Architecture

Layer	Size	Kernel Size	Stride	Padding
Input Image	299 x 299 x 3			
conv		3 x 3 / 32	2	
conv		3 x 3 / 32	1	
conv padded		3 x 3 / 64	1	same
pool		3 x 3 / 64	2	
conv		3 x 3 / 80	1	
conv		3 x 3 / 192	2	
conv		3 x 3 / 288	1	
3_Inception		As in Fig A1 (Appendix)		same
5_Inception		As in Fig A2 (Appendix)		same
2_Inception		As in Fig A3 (Appendix)		same
pool		8 x 8		
logits		logits		
FC: Softmax				

Notes - The kernel size is height x width / number of filters. Batch normalization is used to prevent overfitting the data. FC is a fully connected layer which means all neurons inside this next layer connect to all the neurons of the previous layer. Softmax activation is simply a multinomial logit function.

Lastly, in order to conserve the cardinal nature of our data, instead of using hard labels we use soft labels as the true y values (Diaz & Marathe 2019, Zang *et al.* 2021). In a categorical variable setting, we use hard labels as the y variable. This means if there are 3 categories A, B, C into which

[4] http://download.tensorflow.org/models/image/imagenet/inception-2015-12-05.tgz

we are classifying our image, then an image of 'A' will have the y label as {1,0,0} and an image of a 'C' will have y variable {0,0,1}. In a cardinal variable setting we want to leverage the intrinsic ordering of the labels, so we use soft labels as the y label. What this means is that if there are 3 possible ratings for the commercial {1,2,3}, then an image with an actual commercial rating of 3 will have the y vector defined as -

{f(abs(actual rating-1)), f(abs(actual rating-2)), f(abs(actual rating-3))}

= {f(abs(3-1)), f(abs(3-2)),f(abs(3-3))} = {f(2), f(1), f(0)}, where abs() is absolute difference.

The 'f' function is chosen such that total value across the label vector sums to 1 and 'f' is increasing such that f(2) > f(1) > f(0). We choose a multinomial logit function for this purpose.

More formally we define y as the seven dimensional encoded vector of our ground truth label for a particular instance of rating r_t as:

$$y_i = \frac{e^{-\phi(r_t, r_i)}}{\sum_{k=1}^{7} e^{-\phi(r_t, r_k)}}, \forall\ r_i \in \Omega, \quad \& \ i = 1, 2, .., 7$$

Where $\Omega = r_1, r_2, ..., r_7$ are the 7 cardinal categories and $\phi(r_t, r_i)$ is a metric loss function of absolute difference that penalizes how far the true metric value of r_t is from the rating $r_i \in \Omega$.

It is easy to see that these soft labels allow us to penalize a rating of 1 more strongly as compared to 2, if the real commercial rating of the image is in fact 3.

3.3.2 Keyword Prediction

Our next task is to predict the keywords associated with each image in our sample. For example, an image of a chocolate milkshake could have a list of keywords associated with it such as - cream, milk-shake, fattening, unhealthy etc. We have a sample of 875 images for which we know the list of keywords associated with those images (i.e., they have labelled keywords by MTurkers). We need to predict these words associated with each image. Looking at this list of keywords, we find that there are a total of approximately 8500 unique keyword labels associated with the full set of labelled images. Thus, we have have a high dimensionality problem for our keyword space and the mapping between the unique keywords to the images results in a sparse matrix. In fact, several of these keywords are associated with exactly one image in our labelled data-set. Due to this high dimensionality of the keywords, the labelled data is not sufficient to train a conventional

classification model to predict keywords for all images.

To address this challenge, we use a pre-trained Word2Vec model in combination for Inception Net v3, to create a transfer learning approach to the keyword prediction task. Word2Vec is a popular natural language processing model that maps the words into real valued vectors called word embeddings using a neural network (Mikolov *et al.* 2013). These word-embeddings are encoded in such a way that words that have similar meanings (share a common context) would be closer to each other in the vector space and the words with different meanings would be farther apart. For our prediction exercise we use a Word2Vec model pre-trained on the Google News corpus of text with about 3 million words and phrases. For each word that you supply to this model as input, the output is a 300 dimensional word-embedding vector. We supply all the 8500 unique keywords from our labelled data-set as inputs to the pre-trained Word2Vec model and generate the 300 dimensional word-embedding vector for each of these keywords. Next, for each image we take a dimension-wise average of the word-embedding vectors for all the keywords associated with an image. At the end of this exercise, we obtain a data-set of a 300 dimensional word-embedding associated with each of the 875 images in our data-set.

Next, we reduce the dimensionality of the images in our data-set. Each image is represented as a 299 x 299 x 3 vector in our data, which is a 268,203 dimensional vector. We use the pre-trained Inception Net model described in the previous dimension prediction exercise to generate a lower dimensional image vector. Each 299 x 299 x 3 image vector is provided as input to the Inception v3 model to get a 2048 dimensional vector as an output. At the end of this exercise, we have 2048 dimensional vectors that represent each of the labelled images in our data-set.

Now, we run ridge regressions with the 2048 dimensional image vectors as the independent X variables and each of the 300 dimensional word embedding as the dependent variable (Y). The loss function for the ridge regression is as defined below:

$$L_{ridge}(\hat{\beta}) = \sum_{i=1}^{N}(y_i - x_i\hat{\beta})^2 + \lambda\sum_{j=1}^{m}(\hat{\beta}_j)^2$$

where N corresponds to the 825 images, m corresponds to the 2048 vector dimensions & λ is the regularization penalty.

So in total we run 300 ridge regressions, one each for the 300 dimensions of the word embed-

dings. Once we train this ridge model, we use the weights from this regression to predict the word embedding vectors for all the remaining unlabelled images in our experiment. We then use the Word2Vec model to back-out the set of 20 top keywords with word-embeddings most similar to the predicted vector.

Lastly, we also use Google's Cloud Vision API to predict the keywords for our images. This is a simple process in which the images are supplied to Google's Cloud Vision API software. This program uses a pre-trained model for image recognition and generating labels for the images, which we can then directly download. This provides us with an additional set of keyword labels for all the images in our experiment.

4 Details of the Post-Classification Analysis

After the completion of the training process of our models for Dimension Predictions as well as Keyword Predictions, we carry out post-classification analysis on a held out test set of 50 images to assess the performance of our trained models. This evaluation provides a final unbiased performance measure for our models' fit.

4.1 Dimension Predictions

First we report the post-classification analysis for the dimension prediction exercise. We calculate the performance of the model on a held-out test set of images. In addition, we also train a CNN model with only two convolutional layers to see how a simple CNN without transfer learning performs on our data-set, thereby comparing our model with the transfer learning weights to a simple CNN trained entirely on our training sample. This model provides us a benchmark against which to assess the output of our transfer learning approach. These results are reported in Table 3 below.

Inception v3 is a standard CNN architecture designed by Google which is used for the transfer learning approach in our model, hence we don't describe it in detail here. The architecture of the Simple CNN that we use is described in Table 4. The key reason for keeping this model simple was that, given the small size of the training data-set that we have, it would be difficult to train a more complex model (higher number of effective parameters) as more data would be needed for convergence. As can be seen from the performance comparisons presented in Table 3, the accuracy as well as the hit rate of the model we train using transfer learning is higher than the Simple CNN.

Table 3: Dimension Prediction Performance

	Inception Net		Simple CNN		Naive		Random	
Labels	**Accuracy**	**Hit-Rate**	**Accuracy**	**Hit-Rate**	**Accuracy**	**Hit-Rate**	**Accuracy**	**Hit-Rate**
Abstract	86%	86%	80%	68%	83%	76%	67%	46%
Commercial	84%	76%	74%	58%	72%	54%	61%	32%
Complex	91%	94%	85%	92%	85%	78%	68%	36%
Creative	85%	84%	80%	74%	85%	76%	67%	40%
Emotive	85%	80%	77%	64%	81%	70%	62%	32%
Likeability	93%	100%	86%	82%	87%	88%	64%	36%
Photorealism	81%	70%	65%	42%	53%	34%	62%	36%

Notes - Inception Net is based on the Inception Net v3 algorithm with transfer learning. The Naive Model assigns the mode of each of the dimensions from the training data as the predicted rating for all the test images. The random model assigns a random value chosen from the discrete scale 1-7 as the score for each dimension of each test image. Accuracy is defined as (1 - error), where error is defined as the ratio of the absolute difference between the predicted rating and the true (M-Turkers labelled) rating divided by 6 (Note that the true M-Turker rating is calculated by rounding the mean of the ratings provided by the 5 independent M-Turkers who rated the picture). We are dividing the difference by 6 since the rating scale for each label varies from 1 to 7. Hit-rate gives us the % of cases in which predicted rating deviation from true value was $<= 1$. Both accuracy and hit-rate is calculated on a held-out test data-set with 50 images.

Lastly, instead of treating the ground truth values as discrete variables on a 1-7 scale and handling them as a multi-class classification problem, we use the continuous values of the ground truth labels obtained using a simple average of the ratings for each image, provided by the five independent M-Turkers that labeled the image. Due to the simple average that is performed to obtain the ground truth values (without any rounding of the decimals), these are continuous values ranging between the 1-7 scale. So as a robustness check we train a similar Inception v3 model as per the architecture specified in Table 2 with these continuous ground truth values, and to accommodate the continuous nature of the labels we change the loss function of the final layer to a mean squared error between the true values and the predictions. The performance from this new model is very similar to our original Inception Net model, and we report the accuracy levels for each of the dimensions on the test set in the Appendix Table A1.

4.2 Keyword Prediction

Next, we report the post-classification results for the Keyword Prediction Model. Similar to the previous case, we evaluate the performance on a held-out sample of 50 test images. The results of the performance of the model are presented in Table 5.

We compare the ridge model with other regularization techniques like Elastic Net and Lasso. We see these are not very different in terms of accuracy of the model, although the keyword prediction

Table 4: Simple CNN Architecture

Layer	Size	Kernel Size	Stride	Padding
Input Image	299 x 299 x 3			
Convolution1		3 x 3 / 32	1	Same
ReLU1				
MaxPool1		2 x 2		
Convolution2		2 x 2 / 64	1	Same
ReLU2				
MaxPool2		2 x 2		
FC1				
ReLU3				
Drop-out	drop-out rate is 0.5			
FC2				

Notes - The kernel size is height x width / number of filters. FC is a fully connected layer which means all neurons inside this next layer connect to all the neurons of the previous layer. FC2 has Softmax activation which is simply a multinomial logit function. ReLU is the Rectified Linear Unit Activation function, which takes the max of the input to the activation function and 0. ReLU is generally preferred over a simple sigmoid function as it takes care of the vanishing gradient problem. We use Drop-out regularization to prevent over-fitting the data, which is a risk given the smaller training data size. Lastly we follow batch normalization because it helps train the model faster and has a desirable regularization effect.

Table 5: Keyword Prediction Performance

Model	**Keywords**	**Word Embeddings**	**Absolute Distance (Mean)**	**Absolute Distance (Sum)**	**Cosine Similarity**
Ridge	46%	95%	0.01	169.03	0.81
Elastic net	46%	95%	0.01	169.9	0.80
Lasso	44%	95%	0.01	171.32	0.80
Naive	20%	94%	0.01	196.81	0.74
Random	0%	74%	0.06	826.04	0.02

Notes - The Naive Model assigns a common word embedding value for all images in the test set, which is calculated by taking a dimension-wise average of the vector, across all images. The Random Model assigns word embeddings for each image by picking a random value which varies between the maximum and minimum vector values as observed in the data. Keywords column calculates the percentage of images for which any one of the predicted keywords matches with the M-Turker labelled keywords. For calculating the Word Embeddings column, we take an average across the (1-error) values for each predicted vector dimension, where error is the difference of the predicted value and the true value, divided by the difference between maximum and minimum values of the true vectors (the true vector for each image is calculated by taking a dimension-wise average of the word-embedding vectors for all the keywords associated with an image). Absolute Distance (Mean) is the average of the absolute difference between the predicted and true vector values of the images, across all dimensions. Absolute Distance (Sum) is the sum of the absolute difference between the predicted and true vector values of the images, across all dimensions. Finally cosine similarity is the average across images of a measure of similarity between the predicted word embedding and the true word embedding. Cosine similarity between any two given vectors is defined as the ratio of the dot product of the two vectors and the product of the two vectors' magnitudes. All calculations are based on a held-out test data-set of 50 images.

is clearly the best in Ridge. We also compare our model with two other models which don't use any sophistication to make predictions: the Naive and the Random models. The Naive Model simply outputs a constant prediction. In our case, this common prediction is calculated by taking a dimension-wise simple average of the word-embedding vector, across all training images. On the other hand, the Random Model makes predictions based on uniform random value selections. Therefore for each element of the word-embedding prediction, the Random Model picks a random value which varies uniformly between the maximum and minimum vector values as observed in the training data. We see that our model outperforms both of these across the performance indices.

5 Results

We conduct our analyses through a difference-in-difference framework in order to leverage both the presence of pre-treatment data and the random timing of treatment for treatment group users over the experimental period. The advantage of this framework over a simple means comparison between treatment and control is that we reduce noise by accounting for pre-treatment variation through user-level fixed effects and account precisely for when a user became treated based on their award date. Our generalized specification is as follows:

$$DependentMeasure_{ijt} = \beta_1 \cdot I(TreatmentGroup_i \times PostAward_{ijt}) + \eta_i + \delta_t + \epsilon_{ijt} \qquad (2)$$

In this model, $DependentMeasure_{ijt}$ is our dependent variable measure of interest (which will be either a rating prediction or a distance measure arising from our machine learning models) for the image or project j created by user i at time t. β_1 is our estimated treatment effect, arising from the interaction $I(TreatmentGroup_i \times PostAward_{ijt})$ of $TreatmentGroup_i$: whether the user i is in the treatment group and $PostAward_{ijt}$: whether the image or project j was created t after user i was awarded. We control for persistent differences in creation across users through the use of user fixed effects η_i. These η_i also function as the group identifier in the difference-in-difference framework, while the time-based identifier is the series of fixed effects δ_t which captures week-level common network shocks. We compute robust standard errors and cluster at the project level when

appropriate.

5.1 Analysis - Dimension Labels

We begin our examination with the predefined labels as generated by our transfer learning approach to dimension prediction. For each image, we select the prediction for the each dimension in $\{abstractness, commercialintent, complexity, emotiveness, likeability, photorealism\}$ by calculating the label given by our model as the model predicted level times the confidence in the given level. This gives us a numerical label for each dimension for each image in our dataset, which we feed into our difference-in-difference specification. The results of this analysis is given in Table 6.

Table 6: Treatment Effect Model - Predefined Dimensions

	Dependent variable:						
	Abstract	Commercial	Complex	Creative	Emotive	Likeability	Photorealism
	(1)	(2)	(3)	(4)	(5)	(6)	(7)
TreatmentGroup x PostAward	−0.005	−0.029	0.002	0.001	0.021	0.023	0.051
	(0.017)	(0.039)	(0.027)	(0.026)	(0.026)	(0.022)	(0.057)
Observations	30,803	30,803	30,803	30,803	30,803	30,803	30,803
R^2	0.431	0.491	0.382	0.437	0.461	0.398	0.521
Adjusted R^2	0.419	0.480	0.370	0.426	0.450	0.386	0.512
Residual Std. Error (df = 30198)	0.355	0.687	0.541	0.532	0.522	0.467	1.104

Note: $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$

At least in terms of raw levels, we do not show any significant difference as a result of the award. However, looking at aggregate changes in raw levels has the potential to hide changes in creative behavior, such as if creators have heterogeneous reactions to awards based on the characteristics of the project that was featured.

To address this, we aim to create a measure of similarity between each image a user created either before or after being awarded and the images in the featured project. To do so, we first generate the featured project's average characteristics by taking the mean ratings on each dimension. Then, we normalize ratings utilizing a Z-score to account for differences in mean and variance between dimensions. Finally, we construct three (dis)similarity measures, including cosine similarity, Euclidean distance, and dot product similarity.

The results of this analysis are given in Table 7. Here, we see that artists create significantly different content after receiving an award as compared to before receiving the award. This is shown

in the significant increase in Euclidean distance and a decrease in dot product similarity (note: these quantities are inversely related as one is a measure of difference and the other a measure of similarity).

Table 7: Treatment Effect Model - Dimension Similarity

	Dependent variable:		
	Cosine Similarity	Euclidean Distance	Dot Product
	(1)	(2)	(3)
TreatmentGroup x PostAward	−0.034	0.078*	−0.394**
	(0.021)	(0.047)	(0.172)
Observations	29,246	29,246	29,246
R^2	0.352	0.314	0.512
Adjusted R^2	0.339	0.301	0.503
Residual Std. Error (df = 28681)	0.401	0.950	3.553

*Note: *p<0.1; **p<0.05; ***p<0.01*

5.2 Analysis - Word2Vec Keyword Labels

We conduct an analysis utilizing our predictions from the Word2Vec model. To do so, we generate keyword predictions utilizing a nearest neighbors algorithm to return keyword labels for images based on their representation in the embedded space. We measure the similarity between a given project and the featured project by counting overlapping keywords between the two projects, with a second measure normalizing the count of overlapping keywords by the number of keywords in the featured project. The results of this analysis is shown in Table 8.

These tables replicate our findings that content creators exhibit significantly more novel or different content after being treated.

5.3 Analysis - Pre-trained Model Labels

Finally, we utilize the outputs of pre-trained models in Google's Cloud Vision API. These models trade off the benefits of training on artwork and particularly abstract representations of objects with the benefits of a vastly larger training set. For each experimental user, we create image-wise comparisons of overlapping keyword counts and a fraction of shared keywords (normalized by the

Table 8: Treatment Effect Model - Word2Vec Keyword Labels

	Dependent variable:	
	commonWordCount	commonWordFrac
	(1)	(2)
TreatmentGroup x PostAward	-5.125^{**}	-0.023^{***}
	(2.121)	(0.008)
Observations	3,829	3,829
R^2	0.656	0.489
Adjusted R^2	0.597	0.401
Residual Std. Error (df = 3264)	28.965	0.113

Note: $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$

counts observed in the featured project) for potential pair of images between the user's featured and non-featured projects. Our results, as shown in Table 9, are consistent with our transfer learning approach using Word2Vec. Users in the treatment group create significantly different images after being treated, with an average overlap of 7.2 fewer terms per project. This represents a decrease of 11.4 percentage points in the fraction of overlapping terms.

Table 9: Treatment Effect Model - Pre-trained Google Cloud Vision API Keyword Labels

	Dependent variable:	
	Google commonWordCount	Google commonWordFrac
	(1)	(2)
TreatmentGroup x PostAward	-7.219^{***}	-0.114^{***}
	(2.016)	(0.017)
Observations	95,365	95,365
R^2	0.011	0.025
Adjusted R^2	0.011	0.025
Residual Std. Error (df = 95363)	26.373	0.277

Note: $^{*}p<0.1$; $^{**}p<0.05$; $^{***}p<0.01$

5.4 Heterogeneity in Effects by Level of Prior Recognition

The prior literature suggests that individuals experience satiation of extrinsic motivators such as external recognition, which would suggest that prior levels of external recognition would moderate observed effects of recognition on subsequent content creation. To test this, we examine heterogeneity in our estimates by utilizing a median split by level of prior recognition. Specifically, we divide our sample evenly based on the average daily appreciations the user received in the pre-experimental period. We replicate our prior analyses, first looking at differences induced by experimental assignment in the post treatment period in levels within our pre-defined dimensions, then examining content similarity within the pre-defined dimensions, and then finally keywords from our Keyword Prediction Model. We find that users in the above median prior recognition group exhibit increased content novelty in the post-treatment period as measured both by distance and similarity in latent space (Table 11) and keywords (Table 12), while those with below median prior recognition do not show this change in content production. Replicating our main results, the raw levels do not show any significant difference as a result of the treatment.

These results are consistent with a mechanism of satiation of extrinsic motivation. Creators with lower prior levels of recognition interpret new recognition as a signal of what peers recognize and value, and thus either do not change or even intensify their production of content similar to recognized works in pursuit of further recognition. In contrast, creators with high levels of prior recognition exhibit extrinsic satiation, and therefore can diversify content production to satisfy intrinsic motivations once extrinsic recognition is satisfied, thus taking risks with content production which differs significantly from their prior work.

5.5 Discussion - Stable Unit Treatment Value Assumption

Given that this experiment is conducted on a social network and featured works are visible to all users on the site, there is the potential for spillovers from featured works as a result of the experiment onto other users in the experiment, as these users may also visit the front page of the website and thereby be inspired by the works featured there. These spillovers have the potential to violate the stable unit treatment value assumption (SUTVA) as discussed by Cox (1958) and Rubin (1980), as there would be a dependency between users' observations and the treatment assignment

Table 10: Heterogeneity in Effects by Previous Popularity - Predefined Dimensions

	Dependent variable:						
	Abstract	Commercial	Complex	Creative	Emotive	Likeability	Photorealism
	(1)	(2)	(3)	(4)	(5)	(6)	(7)
TreatmentGroup x PostAward	0.025	−0.032	−0.006	0.028	0.006	0.026	0.017
(Below Median Recognition)	(0.034)	(0.078)	(0.043)	(0.046)	(0.046)	(0.038)	(0.094)
Observations	8,217	8,217	8,217	8,217	8,217	8,217	8,217
R^2	0.474	0.427	0.388	0.449	0.403	0.355	0.577
Adjusted R^2	0.455	0.407	0.366	0.430	0.383	0.333	0.562
Residual Std. Error (df = 7940)	0.353	0.709	0.514	0.513	0.509	0.433	1.048
TreatmentGroup x PostAward	−0.017	−0.025	0.008	−0.009	0.012	0.018	0.081
(Above Median Recognition)	(0.020)	(0.046)	(0.034)	(0.031)	(0.032)	(0.027)	(0.070)
Observations	21,029	21,029	21,029	21,029	21,029	21,029	21,029
R^2	0.410	0.523	0.377	0.432	0.481	0.412	0.502
Adjusted R^2	0.401	0.515	0.367	0.423	0.473	0.402	0.494
Residual Std. Error (df = 20701)	0.355	0.670	0.546	0.530	0.522	0.473	1.122

*Note: *p<0.1; **p<0.05; ***p<0.01*

Table 11: Heterogeneity in Effects by Previous Popularity - Dimension Similarity

	Dependent variable:					
	Below Median Recognition			Above Median Recognition		
	Cosine Similarity	Euclidean Distance	Dot Product Similarity	Cosine Similarity	Euclidean Distance	Dot Product Similarity
	(1)	(2)	(3)	(4)	(5)	(6)
TreatmentGroup x PostAward	−0.012	0.079	−0.384	−0.051**	0.100*	−0.549***
	(0.042)	(0.096)	(0.364)	(0.025)	(0.056)	(0.201)
Observations	8,217	8,217	8,217	21,029	21,029	21,029
R^2	0.358	0.317	0.450	0.355	0.318	0.534
Adjusted R^2	0.336	0.293	0.431	0.345	0.308	0.527
Residual Std. Error	0.411 (df = 7940)	0.954 (df = 7940)	3.395 (df = 7940)	0.396 (df = 20701)	0.946 (df = 20701)	3.605 (df = 20701)

*Note: *p<0.1; **p<0.05; ***p<0.01*

Table 12: Heterogeneity in Effects by Previous Popularity - Keyword Labels

	Dependent variable:			
	Below Median Recognition		Above Median Recognition	
	commonWordCount	commonWordFrac	commonWordCount	commonWordFrac
	(1)	(2)	(3)	(4)
treatment_x_post_feature	−4.675	−0.010	−5.868**	−0.030***
	(4.427)	(0.016)	(2.393)	(0.010)
Observations	1,173	1,173	2,595	2,595
R^2	0.664	0.520	0.666	0.482
Adjusted R^2	0.564	0.377	0.619	0.410
Residual Std. Error	29.681	0.106	28.418	0.114

*Note: *p<0.1; **p<0.05; ***p<0.01*

of other users in the experiment. While we cannot entirely rule out all potential mechanisms for social contagion given the networked nature of the site, we provide some discussion of likely avenues for SUTVA violations and provide discussions and evidence that these are unlikely in our context.

One potential concern could arise if the process of running the experiment caused a change in the quantity, quality, variety, or subject matter of featured works on the site. This change could be noticed by users in the experiment, who might alter their creative activities in response to this perceived change in the feature process. To mitigate this, we specifically designed our experiment so as to not alter the process by which works are regularly featured on the site. Our manipulation works with the pre-existing feature process to randomly rearrange the order in which works queued to be featured actually become featured. There is no increase in the number of featured works as a result of the experiment, and works randomly assigned to either the treatment or control group as a result of the study were drawn from the set of works identified by Behance curators and queued to be featured. From the perception of all users on the site, there is no difference in the number, nature, or content of works featured during the experimental period versus outside the experiment, and therefore neither the control group nor the treatment group before they are treated (featured) can be affected by the presence of this ongoing experiment.

Another potential concern could be the extent to which users' expectations might be altered by our study. In other contexts, if content creators are nominated for an award and the final award recipient is randomized, creator expectations might be subverted or creators may experience shock or disappointment at the unexpected result. The change in expectations could then influence future creative activities. This pathway is not feasible in our context, as users on the site are unaware of whether their works are being examined, identified, and queued to be featured by the curators until the moment the feature award is given. Feature recognition is also exceptionally rare - at the time of the experiment, 9 million projects existed on the site while only a few dozen are featured daily. The overwhelming majority of users on the site will never have a single work become featured. Based on this, we would not expect creators to form strong expectations of getting recognized, nor would a lack of feature recognition have informative value about the quality of one's work. Thus, it is highly unlikely for a SUTVA violation to occur based on changing user expectations as a result of our experiment's presence or manipulation.

Finally, we construct a robustness check to examine whether the characteristics of the works

featured as a result of the experiment have spillover influence on other experimental users' subsequent content creation. Our test utilizes the randomization of the timing of featuring created by the experimental design and proceeds as follows: for each project, P_{ijt}, indexed by i, created by a user j at time t during the experimental period, we find the nearest (temporally) featured works, $F_{i',-j,t-7}$ and $F_{i'',-j,t+7}$, by other users $-j$ that were featured one week before and one week after i's creation date t. We calculate the distances in terms of predefined characteristic dimensions and similarity between P_{ijt} and each of $F_{i',-j,t-7}$ and $F_{i'',-j,t+7}$, and perform a two-sided t-test to determine if there are persistent differences between the similarity of newly created works to existing, publicly featured work or yet-to-be-featured work. The intuition behind this placebo test is that if the characteristics of a featured work has some spillover effect to influence other users' content creation, then newly created works can influence recent features, which are visible and promoted to everyone via the front page of *Behance* while future features that have not yet been promoted mechanically cannot influence the nature of works created temporally prior. The comparison to a future featured experimental project is useful, as this creates a clean comparison between works that are otherwise similar in characteristics due to the randomization in timing and only differ in their visibility and salience as a result of featuring to peer users in the experiment. As additional robustness, we repeat the same test while using a two week window. From Table 13, we see no evidence that the featuring of works created by other artists affects the nature of subsequent creation.

Table 13: Placebo Test of Peer Effects of Feature Recognition

	p-value, two sided t-test:	
	1 Week Comparison Window	2 Week Comparison Window
Euclidean Distance	0.492	0.303
Dot Product	0.314	0.065*
Cosine Similarity	0.527	0.317
Abstract	0.521	0.163
Commercial	0.495	0.275
Complex	0.904	0.514
Creative	0.844	0.404
Emotive	0.627	0.481
Likeability	0.581	0.525
Photorealism	0.971	0.362

*Note: *p<0.1; **p<0.05; ***p<0.01*

6 Conclusion

This study conducts a field experiment on a large art-sharing website to randomly allocate awards and front page featuring in order to estimate changes in creative activity as a result of attention and recognition in social networks. The direction of attention and recognition is one of the key levers available to social networks, which aim to foster diverse creative content on their sites in order to attract users. Additionally, social network platforms are increasingly investing millions of dollars into contracts with top creators to secure exclusive content for the platform, such as Spotify's partnership with Joe Rogan reportedly valued at $200 million (Rosman *et al.* 2022) or TikTok's collaboration with the National Hockey League (TikTok Newsroom 2022). As such, the question of whether awards, attention, and recognition foster greater creativity amongst top creators is a critical one as websites determine how to allocate scarce user attention.

Our analysis employs machine learning algorithms to analyze creative artwork at scale and using parsimonious human-labeled training data and a transfer learning approach. We find that after being featured, creators subsequently create different content from their awarded work. This result is consistent across multiple measures of content similarity, including low-dimensional structured embeddings, medium-dimensional unstructured representations, and pre-trained industrial tools for image labeling.

These results have implications for managers of social networks, who face questions of whether the recognition of top content creators simply encourages creation of repetitive, similar content to the works recognized. We find that the opposite is the case, with non-recognized users actually creating more monotonous content. These results are encouraging, but future research is needed to determine how social network managers can best allocate attention. Expanding the number of works that receive recognition has the potential to dilute the perceived value of the award. Further, it is unclear whether non-top users benefit similarly from award recognition. Finally, much of the value of an award may be in the aspirational value it provides to the large number of non-awardees. These are interesting questions for future research, though may be difficult to assess in large field experiments such as this one.

References

1. Aaltonen, A. & Seiler, S. Cumulative Growth in User-Generated Content Production: Evidence from Wikipedia. *Management Science* **62,** 2054–2069 (2015).
2. Burtch, G., He, Q., Hong, Y. & Lee, D. How do peer awards motivate creative content? Experimental evidence from Reddit. *Management Science* **68,** 3488–3506 (2022).
3. Ciampaglia, G. L., Flammini, A. & Menczer, F. The production of information in the attention economy. *Scientific Reports* **5,** 1–6 (2015).
4. Cornelis, B., Dooms, A., Cornelis, J., Leen, F. & Schelkens, P. *Digital painting analysis, at the cross section of engineering, mathematics and culture* in *2011 19th European Signal Processing Conference* (2011), 1254–1258.
5. Cox, D. R. Planning of experiments. (1958).
6. Diaz, R. & Marathe, A. *Soft labels for ordinal regression* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), 4738–4747.
7. Ederer, F. & Manso, G. Is pay for performance detrimental to innovation? *Management Science* **59,** 1496–1513 (2013).
8. Fei-Fei, L., Deng, J. & Li, K. ImageNet: Constructing a large-scale image database. *Journal of vision* **9,** 1037–1037 (2009).
9. Frey, B. S. & Jegen, R. Motivation crowding theory. *Journal of Economic Surveys* **15,** 589–611 (2001).
10. Gallus, J. Fostering public good contributions with symbolic awards: A large-scale natural field experiment at Wikipedia. *Management Science* **63,** 3999–4015 (2017).
11. Gneezy, U., Meier, S. & Rey-Biel, P. When and why incentives (don't) work to modify behavior. *Journal of Economic Perspectives* **25,** 191–210 (2011).
12. Goude, G. & Derefeldt, G. A study of Wolfflin's system for characterizing art. *Studies in Art Education* **22,** 32–41 (1981).

13. Huang, J. T. & Narayanan, S. Effects of Attention and Recognition on Engagtement, Content Creation and Sharing: Experimental Evidence from an Image-Sharing Social Network. *Working Paper, University of Michigan* (2021).

14. Huotari, P. & Ritala, P. When to switch between subscription-based and ad-sponsored business models: Strategic implications of decreasing content novelty. *Journal of Business Research* **129,** 14–28 (2021).

15. Isen, A. M., Daubman, K. A. & Nowicki, G. P. Positive Affect Facilitates Creative Problem Solving. *Journal of Personality and Social Psychology* **52,** 1122–1131 (1987).

16. Jin, J., Li, Y., Zhong, X. & Zhai, L. Why users contribute knowledge to online communities: An empirical study of an online social Q&A community. *Information & Management* **52,** 840–849 (2015).

17. Kummer, M. E. *Spillovers in Networks of User Generated Content: Evidence from 23 Natural Experiments on Wikipedia* Working paper, University of East Anglia. 2013.

18. Larkin, I. Paying $30,000 for a gold star: An empirical investigation into the value of peer recognition to software salespeople. *Working Paper, UCLA* (2011).

19. Lecoutre, A., Negrevergne, B. & Yger, F. *Recognizing art style automatically in painting with deep learning* in *Asian conference on machine learning* (2017), 327–342.

20. Lerner, J. & Tirole, J. Some Simple Economics of Open Source. *Journal of Industrial Economics* **L,** 197–234 (2002).

21. Mikolov, T., Chen, K., Corrado, G. & Dean, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

22. Muchnik, L., Aral, S. & Taylor, S. J. Social Influence Bias: A Randomized Experiement. *Science* **341,** 647–651 (2013).

23. Negro, G., KovÃ¡cs, B. & Carroll, G. R. Whatâs next? Artistsâ music after Grammy awards. *American Sociological Review* **84,** 644–674 (2022).

24. Rosman, K., Sisario, B., Isaac, M. & Satariano, A. Spotify Bet Big on Joe Rogan. It Got More Than It Counted On. `https://www.nytimes.com/2022/02/17/arts/music/spotify-joe-rogan-misinformation.html` (Feb. 17, 2022).

25. Rubin, D. B. Randomization analysis of experimental data: The Fisher randomization test comment. *Journal of the American statistical association* **75,** 591–593 (1980).

26. Stork, D. G. *Computer vision and computer graphics analysis of paintings and drawings: An introduction to the literature* in *International Conference on Computer Analysis of Images and Patterns* (2009), 9–24.

27. Szegedy, C., Liu, W., *et al. Going deeper with convolutions* in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), 1–9.

28. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J. & Wojna, Z. *Rethinking the inception architecture for computer vision* in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), 2818–2826.

29. Tan, W. R., Chan, C. S., Aguirre, H. E. & Tanaka, K. *Ceci n'est pas une pipe: A deep convolutional network for fine-art paintings classification* in *2016 IEEE international conference on image processing (ICIP)* (2016), 3703–3707.

30. TikTok hits the ice with the NHL and NHLPA to bring hockey fans closer to the action. `https://newsroom.tiktok.com/en-ca/tiktok-nhl-and-nhlpa-partnership` (Feb. 23, 2022).

31. Tinio, P. P. & Gartus, A. Characterizing the emotional response to art beyond pleasure: Correspondence between the emotional characteristics of artworks and viewersâ emotional responses. *Progress in brain research* **237,** 319–342 (2018).

32. Toubia, O. & Stephen, A. T. Intrinsic vs. Image-Related Utility in Social Media: Why Do People Contribute Content to Twitter? *Marketing Science* **32,** 368–392 (2013).

33. Wu, C.-G., Gerlach, J. H. & Young, C. E. An empirical analysis of open source software developers motivations and continuance intentions. *Information & Management* **44,** 253–262 (2007).

34. Wu, F. & Huberman, B. A. Novelty and collective attention. *Proceedings of the National Academy of Sciences* **104,** 17599–17601 (2007).

35. Yosinski, J., Clune, J., Bengio, Y. & Lipson, H. How transferable are features in deep neural networks? *arXiv preprint arXiv:1411.1792* (2014).

36. Zang, H.-X., Su, H., Qi, Y. & Wang, H.-K. *A Compact Soft Ordinal Regression Network for Age Estimation* in *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (2021), 3035–3041.

37. Zhang, X. & Zhu, F. Group Size and Incentives to Contribute: A Natural Experiment at Chinese Wikipedia. *American Economic Review* **101,** 1601–1615 (2011).

38. Zhu, K., Walker, D. & Muchnik, L. Content growth and attention contagion in information networks: Addressing information poverty on Wikipedia. *Information Systems Research* **31,** 491–509 (2020).

39. Zujovic, J., Gandy, L., Friedman, S., Pardo, B. & Pappas, T. N. *Classifying paintings by artistic genre: An analysis of features & classifiers* in *2009 IEEE International Workshop on Multimedia Signal Processing* (2009), 1–5.

Appendix

Figure A1: Labels Generated by Google Vision API

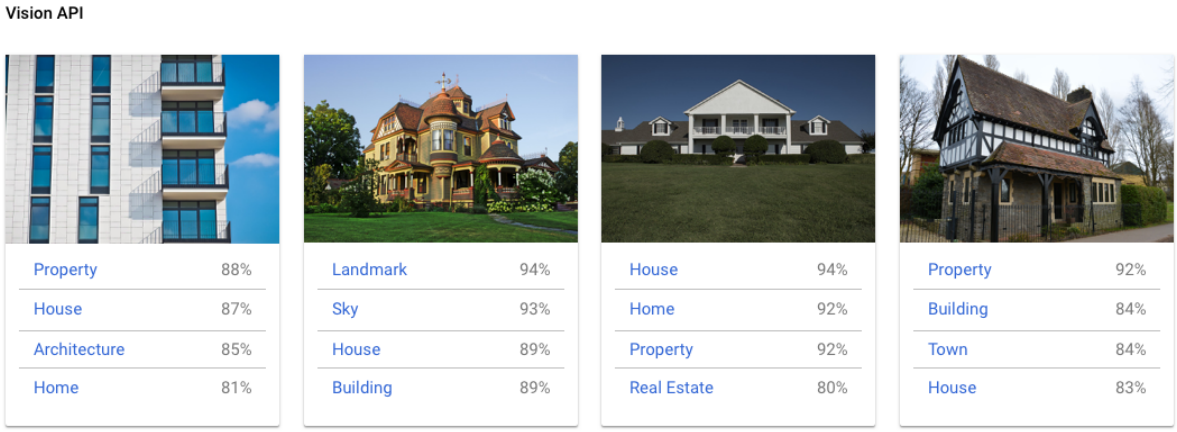


Figure A2: Architecture for Inception Module 1 (as per Szegedy, Vanhoucke, *et al.* 2016)

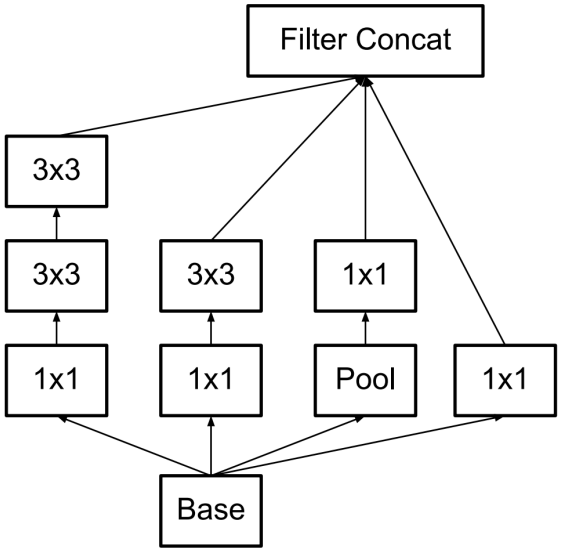


Figure A3: Architecture for Inception Module 2, (as per Szegedy, Vanhoucke, *et al.* 2016, n=7)

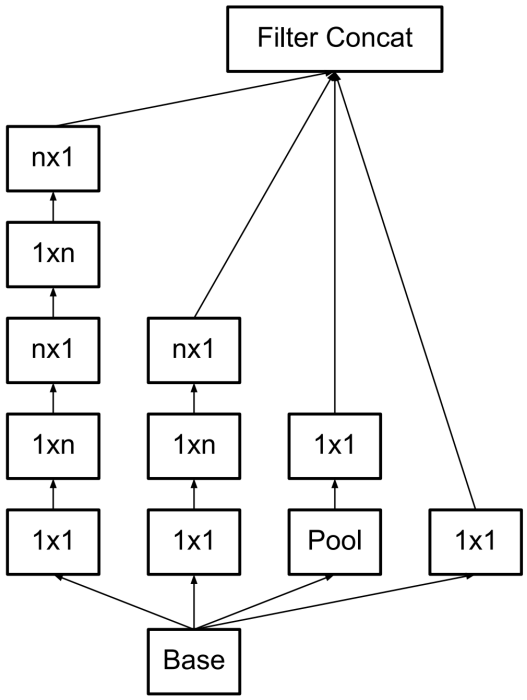


Figure A4: Architecture for Inception Module 3, (as per Szegedy, Vanhoucke, *et al.* 2016)

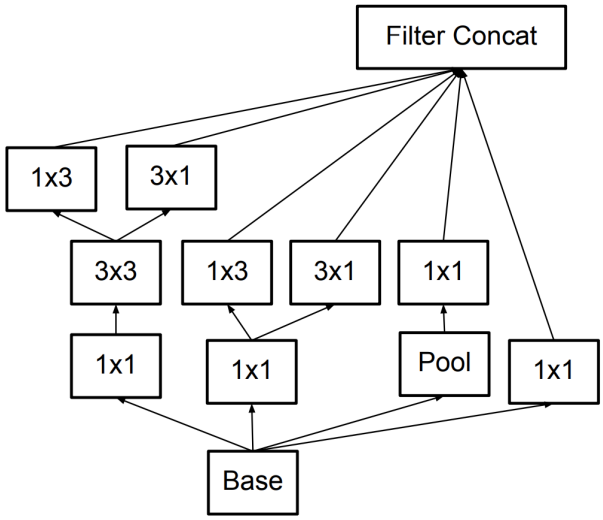


Table A1: Accuracy of Model with Continuous Ground Truth Labels

Labels	**Accuracy Value**
Abstract	87%
Commercial	83%
Complex	88%
Creative	87%
Emotive	85%
Likeability	89%
Photorealism	83%

Notes - This model uses continuous ground truth variable values ranging between 1-7 calculated by taking a simple average (without rounding) across the feature ratings provided by the 5 M-Turkers for each image. The model architecture used is similar to the earlier Inception-v3 based model as defined in Table 2 with the modification being that the last layer of the model is modified to include the MSE (mean squared error) loss function to accommodate the continuous label values. Accuracy is defined as (1 - error), where error is defined as the ratio of the absolute difference between the predicted rating and the true (M-Turker) rating divided by 6 (Note that the true M-Turker rating is calculated by taking an average of the ratings provided by the 5 independent M-Turkers who rated the picture). We are dividing the difference by 6 since the rating scale for each label varies from 1 to 7. The accuracy is calculated on a held-out test data-set with 50 images.