JOURNAL OF
EMERGING INVESTIGATORS

Article

SpottingDiffusion: Using transfer learning to detect Latent Diffusion Model-synthesized images

Muhammad Sahal Mulki[1], Sadaf Adil Mulki[1]
[1] The Westminster School, Dubai, United Arab Emirates

SUMMARY

Differentiating images made by Artificial Intelligence (AI) from real ones has recently become an issue of great importance as realistic AI-generated images rapidly become easier to make and disseminate. Our study aims to present a novel approach for detecting images produced by Latent Diffusion Models (LDMs). The need to detect these images arises when these technologies are used to make images which can mislead the human viewer. We present a solution: an algorithmic way of differentiating images made by LDMs from real ones. In particular, we detail our research on detecting images produced by the Stable Diffusion LDM. We hypothesized that our model could differentiate AI-generated images from real images more accurately than three other recently created methods which aim to do the same. Our transfer learning-based approach to detect LDM-generated images successfully learned to recognize images made by the Stable Diffusion LDM and had a 62.5% accuracy on our testing dataset of LDM-generated images from online, real-world sources. We tested our Transfer Learning-based approach, SpottingDiffusion, against three other recent methods. The best of these methods achieved an accuracy of up to 50.8% on our testing dataset, while SpottingDiffusion achieved an accuracy of 62.5%. These results supported our hypothesis that SpottingDiffusion is more accurate than the other tested methods.

INTRODUCTION

Stable Diffusion was released in August 2022 to realistically generate images using Latent Diffusion Models (LDMs), a type of Artificial Intelligence (AI) model which can make novel images based on a text prompt (1). LDMs are trained by first learning patterns and characteristics of images from a corpus of text-image pairs which are usually scraped from the internet. To create images, they first begin with a starting image, which is made of random noisy pixels, and a text prompt, which describes a desired output image. Second, they iteratively remove the noise and start to create patterns consistent with the text prompt. Eventually, the noisy image results in a satisfactory image that corresponds to the original image prompt. This process is effective at making images that are highly realistic.

Stemming from the numerous hoaxes which have been created with the help of LDMs, public worry and fear has arisen about technologies such as LDMs being used to perpetrate mass disinformation (2, 3). Before the advent of LDMs for image synthesis, the prevailing method to artificially generate images was using the Generative Adversarial Network (GAN) architecture of models (4). Various approaches already exist to detect AI-generated images, but many of these approaches only focus on GAN-generated images (5, 6, 7). Due to LDMs being a relatively new technology for image synthesis, detection of LDM-generated images is greatly understudied. Our research aims to fill this research gap.

We specifically focused on detecting images made by the Stable Diffusion LDM (8). Stable Diffusion is one of the models most at risk for creating obscene images because of its open-source nature and lack of a robust safety filter (9). Another reason we have focused on Stable Diffusion is its widespread use, which means there are an abundance of Stable Diffusion-generated images to collect. Stable Diffusion has over 10 million registered users on its official channels alone and has generated an estimated total of 12.590 billion images (10). Another popular LDM is Midjourney (11). Midjourney is similar to Stable Diffusion in realism and quality and thus is another of the most used LDMs online with 15 million registered users and an estimated total of 964 million images generated (10). Additionally, there is DALL-E 2, developed by OpenAI and released in 2022 (12). DALL-E 2 has more than 1.5 million users, generating an estimated more than 2 million images a day (10).

A certain downfall of these LDMs are the sometimes glaring artifacts and inconsistencies left in from the diffusion process. These artifacts may manifest in prominent ways such as uneven, garbled hands, or sometimes jumbled and warped text in the image. However, they can also manifest in the form of more subtle markers, such as unique patterns held in the noise of an LDM-generated image, which can be extracted using specialized tools and methods, such as noise residual extraction and spectral analysis, as Corvi et al. demonstrated (13). Errors, and inconsistencies, whether minuscule or prominent, can be used to spot an LDM-generated image, and as such, SpottingDiffusion seeks out these artifacts to classify LDM-generated images. To create the SpottingDiffusion model, we used the method of transfer learning. Transfer learning is a process in Machine Learning where a pre-trained “base model” is re-trained or re-applied onto another task from which it was originally meant for (14). In practice, this works because large “base models” such as the one we use, learn basic pattern recognition which can be reapplied to a whole variety of tasks for which the model may not have been originally meant (e.g. detecting LDM-generated images). We hypothesized that training our model, SpottingDiffusion, with better and more relevant data than what the three methods we compared it against used, would improve its accuracy in detecting AI-generated images

https://doi.org/10.59720/23-256

compared to said methods, when tested on a dataset of LDM-generated images found online. SpottingDiffusion achieved an accuracy higher than the other methods, supporting our hypothesis (5, 13, 15). Our new method could improve upon the existing literature in the field of detecting LDM-generated images and allow for more reliable attribution of LDM-generated images.

RESULTS

SpottingDiffusion is a novel approach developed to detect images synthesized by Latent Diffusion Models (LDMs) using transfer learning techniques. The core idea behind SpottingDiffusion is to create a model using transfer learning and use it to detect LDM-generated images. Transfer learning, in our case, consists of leveraging a pre-trained "base model" and fine-tuning it to detect LDM-generated images. This process involves training a classifier on a dataset comprising of both real and LDM-generated images, allowing the model to learn distinct features that help in the identification of LDM-generated images.

We compiled 2 datasets to train, test, and validate SpottingDiffusion; each dataset contained AI-generated and real images. Dataset 1 was used to train and validate SpottingDiffusion. It consisted of 10,000 images, 5,048 real and 4,952 generated by AI (Stable Diffusion). Dataset 1 was further split into two subsets, training and validation. Fifteen percent of Dataset 1 (1,500 images) was used to validate the performance of SpottingDiffusion on unseen images, while the other 85% (7,500 images) was used to train the model. Dataset 2 consisted of 120 images, 60 real and 60 generated by AI (Stable Diffusion, DALL E 2, and Midjourney). Dataset 2 was made up of real and AI-generated images found on online forums (Reddit forums "r/stablediffusion", "r/dalle2" and "r/midjourney" for LDM-generated images, and "r/pics" for real images). SpottingDiffusion gave a training accuracy of 83.13%, a validation accuracy of 77.79% on Dataset 1, and a testing accuracy of 62.5% on Dataset 2. We may further put the results of SpottingDiffusion in context with other approaches to detect synthetically-generated images. We took three other recently created models to detect synthetically-generated images and evaluated their performances on the testing subset. The models we compared SpottingDiffusion to were "Grag2021 retrained on ADM-generated images" from Corvi et al. with an accuracy of 51%, "[Blur+JPEG(0.1)]" from Wang et al. with an accuracy of 50%, and "ResNet50-NoDown-StyleGAN2" from Gragnaniello et al. with an accuracy of 50% (5, 13, 15). SpottingDiffusion outperformed these other models on the testing subset with an accuracy of 63% **(Figure 1)**.

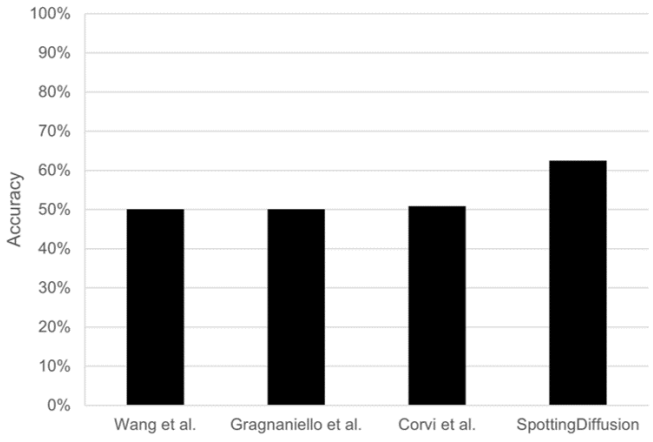

Figure 1: Accuracy of SpottingDiffusion and three pre-existing methods in detecting AI-synthesized images. Accuracy of SpottingDiffusion on a dataset, compared to other methods. All four models were tested on the testing dataset, consisting of real-world LDM-synthesized images and real images found online.

Dataset	Mean Aesthetic Score
Dataset 1, Training and Validation (only LDM-generated images)	3.4
Dataset 1, Training and Validation (only Real images)	3.6
Dataset 2, Testing (only LDM-generated images)	6.6
Dataset 2, Testing (only Real images)	5.6

Table 1: Mean predicted aesthetic scores of all datasets, divided by real vs. LDM-generated images. Predicted aesthetic score values generated by the Aesthetics V1 tool by the LAION foundation.

Furthermore, we analyzed the aesthetic quality of the training and testing images using a tool provided online by the LAION foundation to predict the aesthetics of an image, or how aesthetic it would be rated by a human viewer, on a scale from 1 to 10 (16). We computed the aesthetic scores of each image in all our datasets and averaged these scores to get a mean aesthetic score for each dataset **(Table 1)**.

The mean aesthetic score of real images in Dataset 2 was 5.6, while that for LDM-generated images was 6.6. Statistically, there was a positive correlation (R = +0.43, 95% CI [0.27, 0.57]) between an image in Dataset 2 being LDM-generated and its aesthetic score **(Figure 2)**.

DISCUSSION

In this study, we trained our model, SpottingDiffusion, to detect AI-generated images more accurately than existing approaches. As expected, SpottingDiffusion was most effective at detecting images on which it had already been trained. New, unseen images (validation and testing subsets), were more difficult to classify. SpottingDiffusion had accuracies of 62% and above on testing and validation subsets. SpottingDiffusion also showed an ability to generalize onto detecting images made by models other than Stable Diffusion, namely the LDM models DALL-E 2 and Midjourney (11). This is demonstrated by the accuracy of SpottingDiffusion on the testing subset, which consisted of images made by Stable Diffusion, Midjourney, and DALL-E 2 (8, 11, 12).

We also conducted a statistical analysis of Dataset 2, analyzing the correlation between an image being fake and its aesthetics score. We found that, in Dataset 2, which consisted of real and fake images found on online forums, it was more likely that an image would have an higher aesthetics score if it was fake (R = +0.43, 95% CI [0.27, 0.57]) **(Figure 2).** Interestingly, the correlation between fakeness and aesthetic score in Dataset 2, suggests that AI-generated images are, on average, more aesthetic and better looking than real photos. This would seem to be supported by the fact that some versions of Stable Diffusion (e.g. Stable Diffusion

https://doi.org/10.59720/23-256

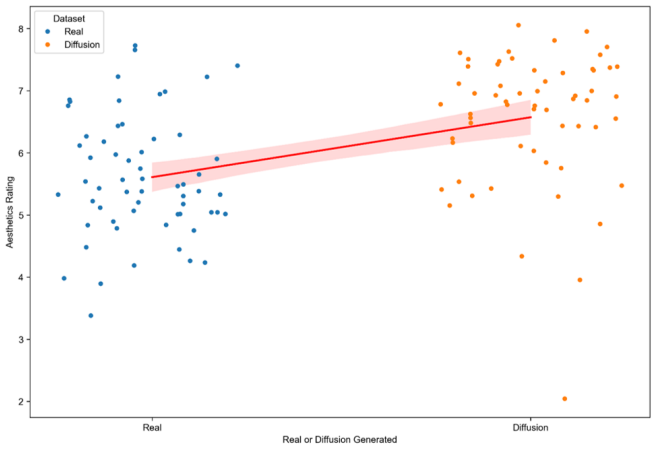

Figure 2: Predicted aesthetic scores of images in Dataset 2. Predicted aesthetics scores of real and fake images in Dataset 2, confidence interval for line of best fit shown in shaded area. Predicted aesthetic score values generated by the Aesthetics V1 tool by the LAION foundation. Jitter plot created using Python, Seaborn, and Matplotlib libraries with Jitter value set to 0.25. Positive correlation (R = +0.43, 95% CI [0.27, 0.57]) between an image in Dataset 2 being LDM-generated and its aesthetic score.

v1-4) were trained on datasets filtered to include only high-aesthetic images (17). The theoretical applications of an LDM-detecting model such as SpottingDiffusion are numerous. For example: i) moderating online art forums to verify if a piece of art is made by a human or is AI-generated and being passed off as human-made, ii) keeping AI-generated images out of datasets that are used to train AI models to make images, and iii) verifying the authenticity and origin of images uploaded on social media platforms.

The accuracy of SpottingDiffusion points to the fact that it has learned how to spot an LDM-generated image. We hypothesized that SpottingDiffusion will outperform the levels of accuracy that other methods in the field have so far achieved. This hypothesis was supported by the difference between our method's accuracy and the accuracies of other methods to detect AI-generated images. SpottingDiffusion differs from existing methods in one major area: the training data used. The detectors we compared SpottingDiffusion to were "Grag2021 retrained on ADM-generated images" from Corvi et al., "[Blur+JPEG(0.1)]" from Wang et al., and "ResNet50-NoDown-StyleGAN2" from Gragnaniello et al (5, 13, 15). Out of these three, Corvi et al., was the only one trained on images produced by LDMs (13). All the other detectors were trained to detect GAN-generated images. As shown, most of the prior methods used images generated by GANs for training, however, we used LDM-generated images for training our model, as we deemed these more relevant. We believe that our model outperformed the others due to this difference in training data. Additionally, we also used the same underlying model architecture that all the three compared methods did: transfer learning.

One limitation of our method is that it focuses solely on detecting LDM-generated images and cannot detect images generated from other architectures. For example, other types of AI-generated images also exist, such as ones that are generated by GANs, but these other types of images cannot be detected by SpottingDiffusion. Additionally, if a new advancement in the field of AI-generated images made LDMs obsolete, it is unlikely that SpottingDiffusion, with the data it has been trained on currently will be able to detect this new type of image generated using a post-LDM architecture. This idea is supported by how the models made to detect GAN-generated images which we tested failed to accurately detect modern LDM-generated images, which they were never trained on. Furthermore, our model cannot directly attribute the generation of an image to one specific LDM: it can only classify an image as being created using an LDM or not. Additionally, the accuracy we achieved on our testing dataset (62%) is currently too low to be used in critical applications (e.g., criminal investigations hinging on potentially faked evidence or confirming the authenticity of a video of an influential figure making controversial remarks). Considering all these limitations, SpottingDiffusion's architecture could be built upon and improved by other future studies to further the field of detecting AI-generated images. We found that SpottingDiffusion could successfully generalize onto detecting LDMs it had not been trained on, namely Midjourney and DALL-E 2 (11, 12).

Future studies building off SpottingDiffusion could use frameworks other than Transfer Learning for constructing the model. For example, future studies could use Vision Transformers (ViT) models to replace MobileNetV2, which was the base model we used in our framework (18, 19). In studies, ViTs have been found to provide equal or better performance compared to Convolutional Neural Networks such as MobileNetV2 (20). The SpottingDiffusion framework could also be used for other similar tasks in detecting content made by AI (audio, video deepfakes, other types of AI-generated images, etc.) and could be generalized for use in other fields.

MATERIALS AND METHODS

Making the datasets

Dataset 1 was compiled for training and validating the model and consisted of 10,000 images in total. 4,952 out of those 10,000 images were Stable Diffusion-generated images and the other 5,048 were real images. Fifteen percent of Dataset 1 was taken as a validation subset for estimating our model's abilities on unseen data during training. The other 85% of Dataset 1 was used for training our model, and this was labeled as the training subset. The training subset consisted of 8,500 images, while the validation subset had 1,500. To make this main dataset for training and validating SpottingDiffusion, we looked to 3 main sources. We used the LAION-400M dataset of images and the LAION-aesthetics-4.75plus dataset for the Real Images class in the dataset. And finally, for the "LDM-generated Images" class of the dataset, we used Lexica, a website containing a large collection of images made by Stable Diffusion (21, 22, 23). We scraped and subsequently downloaded 4,952 Stable Diffusion-generated images from Lexica and 5,048 real images for our dataset from LAION-400M and LAION-aesthetics-4.75plus, making the total number of images in the dataset 10,000. We computed the correlation coefficient, and trendline between aesthetics and realness values using the Python "SciPy", "Matplotlib" and "seaborn" libraries (24, 25, 26).

We also compiled a secondary testing subset (Dataset 2) of real images and images made by the Stable Diffusion, DALL-E 2, and Midjourney LDMs (8, 11, 21). The purpose of this secondary dataset was testing the model on real-world

https://doi.org/10.59720/23-256

examples. To do this, we scraped the 20 most popular AI-made photos from the Reddit forums "r/stablediffusion", "r/dalle2" and "r/midjourney", for the Stable Diffusion, DALL-E 2 and, Midjourney LDMs, respectively . Furthermore, for the real class of photos we scraped the 60 most popular photos from the Reddit forum "r/pics". This resulted in a dataset of 120 images, 60 of which were real, and 60 of which were synthetic. The 60 synthetic photos were comprised of 20 photos from each LDM listed above. This entire dataset was used as a testing subset. The Dataset 2/testing subset, and the validation subset of dataset 1 were completely distinct from each other.

We compared the performance of SpottingDiffusion with 3 other recently published methods to detect AI-generated images. The detectors we took were "Grag2021 retrained on ADM-generated images" from Corvi et al., "[Blur+JPEG(0.1)]" from Wang et al., and "ResNet50-NoDown-StyleGAN2" from Gragnaniello et al (5, 13, 15). Out of these three detectors, Corvi et al. was the only one trained on images produced by LDMs. All the other detectors were trained to detect GAN-generated images.

Making the model

We used the TensorFlow (27) Python library to implement and train the SpottingDiffusion model **(Figure 3)**. For constructing the SpottingDiffusion model, we used the method of transfer learning. We used the pre-trained model MobileNetV2 as the "base model" and used model weights pre-trained on the "ImageNet" dataset for the MobileNetV2 network (19). We froze the pre-trained MobileNetV2 model before training, i.e., it was not further trained on our dataset. We subsequently used a "classification head" model which processes the output from MobileNetV2. The "classification head" consisted of a "dropout" layer with a rate of 0.3 and a "fully-connected layer" consisting of 256 neurons or units to transform the output from MobileNetV2 into a final prediction of an image's authenticity (28). The dropout layer introduces noise into the model layers by not training some randomly selected neurons at all. This has the effect of making the model more robust and allows it to generalize from the training data more effectively. As we used a dropout rate of 0.3, 30% of data coming in from MobileNetV2 to the "classification head" was randomly dropped during training. Crucially, dropout was not applied when the model was run for inference after the training was complete.

We used an image input shape of 256x256 pixels for our model because this is less computationally intensive than a larger input shape would be and thus ensures that our model can run natively in computationally constrained scenarios (e.g. on a mobile phone). Finally, our model was trained on the training subset for 12 epochs, with a learning rate of 0.00001 to obtain the results presented in this paper. At the end of the training, we evaluated the model on the validation, training, and testing subsets and obtained the results that were presented within the Results section.

GitHub Repository

The GitHub Repository for SpottingDiffusion can be found at www.github.com/sahal-mulki/SpottingDiffusion.

Training and validation subsets

The training and validation subsets of the dataset can be found at www.kaggle.com/datasets/sahalmulki/stable-diffusion-generated-images, and at www.kaggle.com/datasets/sahalmulki/spottingdiffusion-testing-dataset for the testing subset.

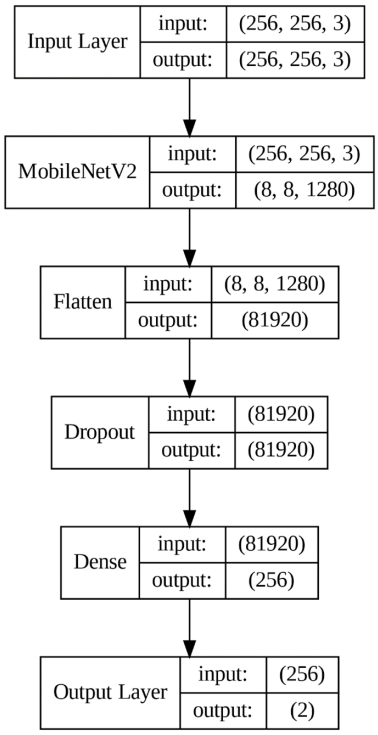

Figure 3: The internal structure of SpottingDiffusion. Overview of how SpottingDiffusion processes and moves information using the "tf.keras.utils.plot_model" utility in TensorFlow. The numbers in brackets represent the array sizes at each layer's input or output. For example, "(256, 256, 3)" represents an image with a height and width of 256 pixels and three color channels (red, green, and blue). The output "(2)" signifies a one-dimensional array (list) with 2 values representing the probabilities of the image being either real or generated by LDM.

ACKNOWLEDGEMENTS

We would like to thank Erfan Firouzi of "The Wildlife Focus" and Muhammad Adil Mulki for their invaluable insights and feedback into the writing of this article.

Received: September 30, 2023
Accepted: May 19, 2024
Published: November 15, 2024

REFERENCES

1. Rombach, Robin, et al. "High-Resolution Image Synthesis with Latent Diffusion Models." *arXiv E-Prints*, 13 Apr. 2022, pp. arXiv:2112.10752, https://doi.org/10.48550/arXiv.2112.10752.
2. Alba, Davey. "How Fake AI Photo of a Pentagon Blast Went Viral and Briefly Spooked Stocks." *Bloomberg.Com*, 22 May 2023. https://www.bloomberg.com/news/articles/2023-05-22/fake-ai-photo-of-pentagon-blast-goes-viral-trips-stocks-briefly.
3. Stanley-Becker, Isaac, et al. "Fake Images of Trump Arrest Show 'Giant Step' for AI's Disruptive Power." *Washington Post*, 24 Mar. 2023. https://www.washingtonpost.com/

https://doi.org/10.59720/23-256

politics/2023/03/22/trump-arrest-deepfakes/.
4. Goodfellow, Ian J., et al. "Generative Adversarial Networks." *arXiv E-Prints*, June 2014, pp. arXiv:1406.2661, https://doi.org/10.48550/arXiv.1406.2661.
5. Wang, Sheng-Yu, et al. "CNN-Generated Images Are Surprisingly Easy to Spot... for Now." *arXiv E-Prints*, 4 Apr. 2020, pp. arXiv:1912.11035, https://doi.org/10.48550/arXiv.1912.11035.
6. Marra, Francesco, et al. "Detection of GAN-Generated Fake Images over Social Networks." *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, IEEE, 2018, pp. 384–89. https://doi.org/10.1109/MIPR.2018.00084.
7. Tang, Guihua, et al. "Detection of GAN-Synthesized Image Based on Discrete Wavelet Transform." *Security and Communication Networks*, edited by Beijing Chen, vol. 2021, June 2021, pp. 1–10. https://doi.org/10.1155/2021/5511435.
8. Rombach, Robin, et al. *CompVis/Stable-Diffusion: A Latent Text-to-Image Diffusion Model*. https://github.com/CompVis/stable-diffusion. Accessed 1 Apr. 2023.
9. Rando, Javier, et al. "Red-Teaming the Stable Diffusion Safety Filter." *arXiv E-Prints*, 10 Nov. 2022, pp. arXiv:2210.04610, https://doi.org/10.48550/arXiv.2210.04610.
10. *AI Image Statistics: How Much Content Was Created by AI*. 15 Aug. 2023, https://journal.everypixel.com/ai-image-statistics.
11. *Midjourney*. https://www.midjourney.com/home/?callbackUrl=%2Fapp%2F. Accessed 1 Apr. 2023.
12. Ramesh, Aditya, et al. "Hierarchical Text-Conditional Image Generation with CLIP Latents." *arXiv E-Prints*, 12 Apr. 2022, pp. arXiv:2204.06125, https://doi.org/10.48550/arXiv.2204.06125.
13. Corvi, Riccardo, et al. "On the Detection of Synthetic Images Generated by Diffusion Models." *arXiv E-Prints*, 1 Nov. 2022, pp. arXiv:2211.00680, https://doi.org/10.48550/arXiv.2211.00680.
14. Bozinovski, Stevo. "Reminder of the First Paper on Transfer Learning in Neural Networks, 1976." *Informatica*, vol. 44, no. 3, Sept. 2020. https://doi.org/10.31449/inf.v44i3.2828.
15. Gragnaniello, D., et al. "Are GAN Generated Images Easy to Detect? A Critical Analysis of the State-Of-The-Art." *2021 IEEE International Conference on Multimedia and Expo (ICME)*, July 2021, pp. 1–6. https://doi.org/10.1109/ICME51207.2021.9428429.
16. *LAION-AI/Aesthetic-Predictor*. 2022. LAION AI, 14 Feb. 2024. *GitHub*, https://github.com/LAION-AI/aesthetic-predictor.
17. *CompVis/Stable-Diffusion-v1-4 · Hugging Face*. https://huggingface.co/CompVis/stable-diffusion-v1-4. Accessed 16 July 2024.
18. Dosovitskiy, Alexey, et al. "An Image Is Worth 16x16 Words: Transformers for Image Recognition at Scale." *arXiv E-Prints*, 3 June 2021, pp. arXiv:2010.11929, https://doi.org/10.48550/arXiv.2010.11929.
19. Sandler, Mark, et al. "MobileNetV2: Inverted Residuals and Linear Bottlenecks." *arXiv E-Prints*, Jan. 2018, pp. arXiv:1801.04381, https://doi.org/10.48550/arXiv.1801.04381.
20. Li, Yanyu, et al. "Rethinking Vision Transformers for MobileNet Size and Speed." *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, IEEE, 2023, pp. 16843–54. https://doi.org/10.1109/ICCV51070.2023.01549.
21. Schuhmann, Christoph, et al. "LAION-400M: Open Dataset of CLIP-Filtered 400 Million Image-Text Pairs." *arXiv E-Prints*, Nov. 2021, pp. arXiv:2111.02114, https://doi.org/10.48550/arXiv.2111.02114.
22. Schumann, Cristoph. *ChristophSchuhmann/Improved_aesthetics_4.75plus · Datasets at Hugging Face*. https://huggingface.co/datasets/ChristophSchuhmann/improved_aesthetics_4.75plus. Accessed 1 Apr. 2023.
23. *Lexica*. https://lexica.art/. Accessed 1 Apr. 2023.
24. Virtanen, Pauli, et al. "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python." *Nature Methods*, vol. 17, no. 3, Mar. 2020, pp. 261–72. https://doi.org/10.1038/s41592-019-0686-2.
25. Waskom, Michael L. "Seaborn: Statistical Data Visualization." *Journal of Open Source Software*, vol. 6, no. 60, Apr. 2021, pp. 3021. https://doi.org/10.21105/joss.03021.
26. Hunter, John D. "Matplotlib: A 2D Graphics Environment." *Computing in Science & Engineering*, vol. 9, no. 3, May 2007, pp. 90–95. https://doi.org/10.1109/MCSE.2007.55.
27. Abadi, Martın, et al. *TensorFlow: A System for Large-Scale Machine Learning*.
28. Srivastava, Nitish, et al. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting." *Journal of Machine Learning Research*, vol. 15, no. 56, 2014, pp. 1929–58.