**Conformal Alignment: Knowing When to Trust Foundation Models with Guarantees**

**Paper by: Yu Gui, Ying Jin , and Zhimei Ren (2024)  
Project by: Ofek Gottlieb & Saar Tzour-Shaday**

# Abstratct

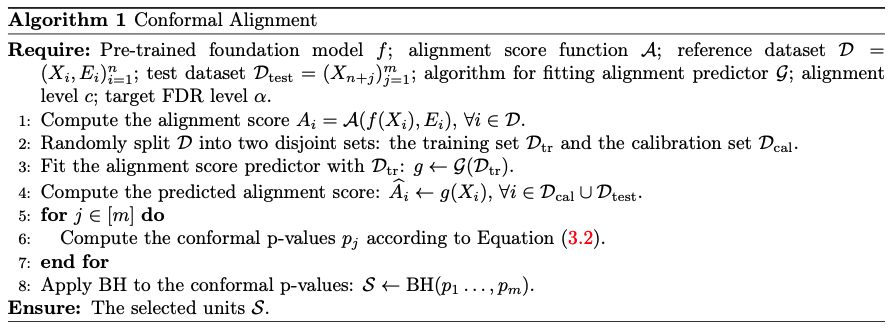
# The paper by Yu Gui et al. (2024) introduces Conformal Alignment, a method that certifies aligned outputs of pre-trained foundation models in generating outputs for various tasks, with strict control over the false discovery rate (FDR). Extending this framework to the art domain, we explored whether a generative model like DALL-E can reliably generate art pieces in the style of famous artists. We assessed the quality of model-generated reproductions of existing paintings using Conformal Alignment.

# Section 1: Background and Problem Setup

The paper “Conformal Alignment: Knowing When to Trust Foundation Models with Guarantees” by Yu Gui et. al (2024) discusses the impressive capabilities of large-scale, pre-trained foundation models in generating various types of outputs for different tasks. However, these models often face issues like factual errors, hallucinations, and bias, which are especially concerning in high-stakes situations. In these careful tasks, the model’s outputs must align with human evaluations before their use. While trusting a foundation model directly for such tasks is out of the question, building multiple testing procedures on its outputs are necessary to guarantee reliability. One of the most desired and practical properties to assure is the ability to control the type I error: the expected proportion of erroneously rejected hypotheses among the rejected ones, which is the focus of the paper.

# Section 2: The Conformal Alignment Procedure, Results and Limitations

After we understand the problem, let’s dive right into the paper’s main contribution and analyze the results in detail. Conformal Alignment is a flexible, effective framework to select those aligned outputs, combining conformal prediction and hypothesis testing:



It begins by computing the ground-truth alignment score for each sample in the reference dataset , followed by splitting it into training set and calibration set . The alignment score predictor model is then trained on the training set and computes the predicted alignment score to each sample in the training and the calibration sets. The method calculates conformal p-values for each test unit , indicating how likely it is that the output is aligned, according to the conformal p-value formula:

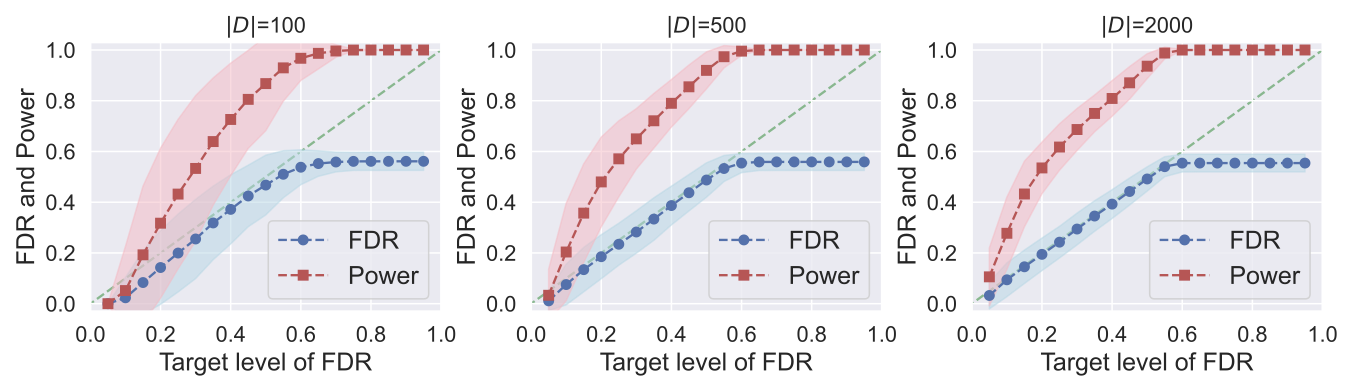
The p-value corresponds to the following null hypothesis: . Rejecting reflects evidence that the (true) alignment score of unit is above the threshold , and therefore the generated output is aligned. The threshold of p-values determined by the Benjamini-Hochberg procedure, as we saw in lecture 10: let denote the ordered statistics of the conformal p-values, the rejection set is then , where .

Overall, this framework aims to optimize the power, i.e. the proportion of selected aligned units out of all aligned test units: , while strictly enforce the FDR constraint, i.e. the proportion of selected units that are not aligned:

##### Results Analysis

The paper suggests 2 domains to experiment their framework. One is the setting of generated radiology reports based on chest X-ray scans. Given a stream of reports, the goal is to select the most correct and aligned to human report, such that the proportion of false positives, i.e. selected reports that are not truly aligned, will be strictly controlled below a given . The other domain is a question anwering task of LLMs, in which alignment means to select only correct answers.

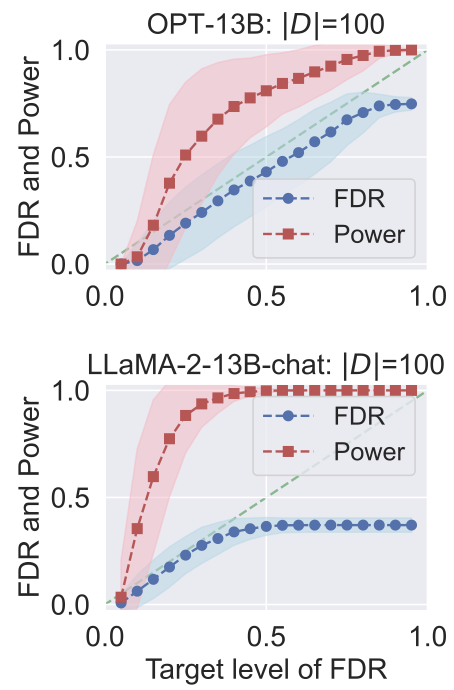
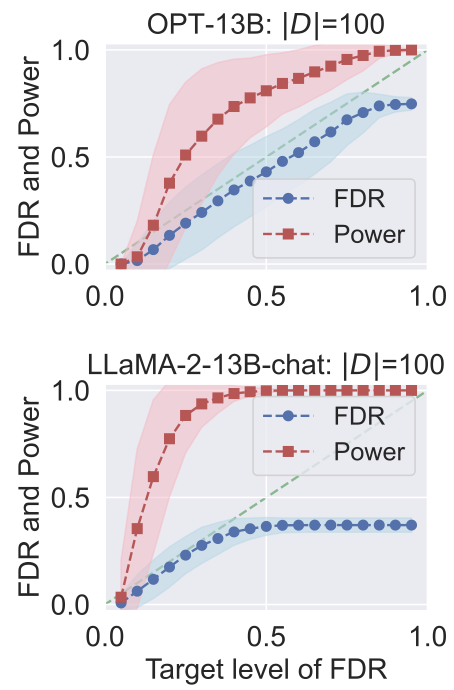
The results demonstrate that Conformal Alignment effectively controls the FDR, showing consistent error control even as the FDR target level increases. For example, Power vs. FDR levels of Conformal Alignment applied on the generated radiology reports of a pre-trained Vision-Transformer and GPT2 encoder-decoder model (figure 6):



Over 500 independent runs, the dots denote the expectancy, and the shaded area denotes the standard deviation. Each plot represents a different sample size of the reference set used in the algorithm. It is noticeable that a few hundred high-quality samples are generally sufficient for effective procedure (500 on average), with larger sample sizes offering better stability in selection, demonstrating reduced variance in both FDR and Power.

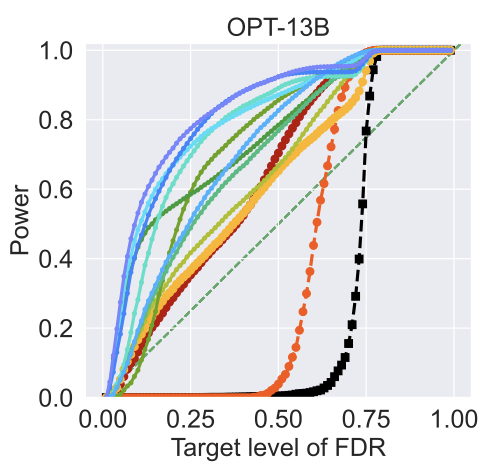
Interestingly, the Power and the FDR curves converge simultaneously, i.e. for the same level of ; it’s no coincidence. When the FDR becomes stably fixed, along with the Power that converges to 1, it means that all the test units were selected. It makes sense because we can’t make more mistakes than the number of not aligned test samples exist.

The writers also observed that more powerful model enables more powerful selection, especially when applying small values of . For example, the question answering domain, they compared the results of the OPT-13B foundation model vs. the LlaMA-2-13B-chat (figure 3):



Clearly, the LlaMA-2-13B-chat is more powerful than OPT-13B: for the Power of the latter is almost 1, while the former results in slightly above 0.5.

In addition to the capability of the foundation model, the features used to train also play a vital role in the power of selection. The writers divided them into 3 subgroups: **Self-evaluation likelihood** – the confidence level of on its own generated outputs. **Input certainty scores** – the variability and uncertainty present in the inputs to the model. **Output confidence scores** – the consistency and confidence of the model's outputs based on their interrelationships. All the features for both domains in the paper were semantic and lexical properties of text using similarity measurements and LLMs, for example rouge-L similarity, the eigenvalues of the graph Laplacian of words embeddings, the pairwise distance based on a degree matrix, etc.

To test the informativesness of each feature and its effect on the Power curve, they conducted the same experiment but with a single feature at a time. In the right are the results on the question answering domain (figure 5). Each curve represents a feature. For the most effective features the Power quickly increases, wheras for the less informative features, no test units were selcted for , but when , the Power increases rapidly to 1, since all test units can be selected without violating the FDR constraint (as we saw in the previous plot, the number of not aligned test samples is 0.4-0.5).

##### Limitations

First of all, Conformal Alignment fits only on a large set of test units, as in multiple hypothesis testing. Therefore, for several tasks, we can’t provide an imidiate decision. For example, a doctor who wishes to use a foundation model to generate a radiology report of an X-ray and test its reliability, must first collect another 400-500 scans before he can apply Conformal Alignment.

Secondly, the provided guerantee is only by expectancy: It’s a game of chance. When using Conformal Alignment in practice with a single test set, it could return a selected set in which the FDR constraint really holds, but it might as well select only false positives. However, if we conduct continuous experiments as researchers, without employing the outputs for a practical use, it can help us evaluate the reliability of the foundation model in general.

Moreover, the framework requires to implement an algorithm that assigns a true alignment score to the reference dataset. Even though computer science research evolve with every passing day, sometimes it’s just not possible for a computer to learn alignment to human values accurately. Even more challenging is engineer the features to train . There are tasks that even humans can’t deduce alignment without a reference. We met these limitations in our domain, which we will discuss further in the next sections.

# Section 3: Creative Extension

For the creative extension, we chose to implement the Conformal Alignment framework on another domain. While it was tempting to choose a truly high-stake task that might really change the world, we decided to choose a field that we are more familiar with, and frankly more fun: Art.

We start with a short background: the golden age of the most inspiring artists such as Van-Gogh and Rembrandt is buried in the 19th century, 20th at most. With the recent emerging of multimodal foundation models such as text-to-image generative models, we were wondering: can we generate more of Van-Goghs fine paintings?

This task was examined long before this project, but as far as we know, no one ever tried to evaluate the quality of such generated paintings. Enthusiastic and impassioned, we formulated our goal: can we trust a model to generate a reliable art piece of the artist by our choice? To fit the domain to the Conformal Alignment framework, we focused on a slightly relaxed version of the task: instead of generating new paintings, we assessed the model’s ability to generate existing ones – those who have reference we can compare the generated paintings with.

##### Pre-trained foundation model

We chose DALL-E3 as our foundation model. DALL-E is a text-to-image model developed by OpenAI using deep learning methodologies to generate digital images from natural language descriptions known as "prompts".

We chose DALL-E because it’s considered the state of the art for this task, aside with Midjourney nd Stable Diffusion. It can generate imagery in multiple styles, including photorealistic imagery and paintings. It can manipulate and rearrange objects in its images and correctly place design elements in novel compositions without explicit instruction.

##### Kaggle (@kaggle) / XDataset creation

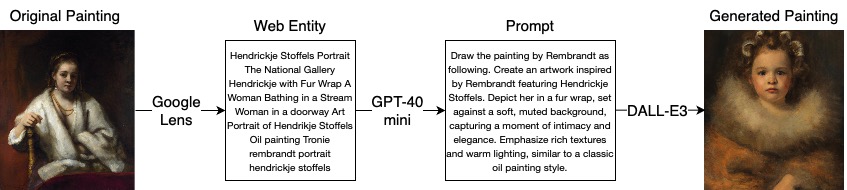
One of the most challenging parts in our project was collecting the dataset. We needed at least 1000 paintings, each tagged with the piece name, artist name, and a prompt describing it. We started by downloading the Kaggle dataset “Best Artworks of All Time – Collection of Paintings of the 50 Most Influential Artists of All Time”.

We focused on 10 famous artists: Vincent van Gogh, Edgar Degas, Pablo Picasso, Pierre-Auguste Renoir, Albrecht Dürer, Paul Gauguin, Francisco Goya, Rembrandt, Alfred Sisley and Titian (Tiziano Vecellio).

Unfortunately, the paintings had only artists tags, without names or description. To cope, we took advantage of Google Lens, a deep learning model developed by Google, designed to bring up relevant information related to objects it identifies. For each painting, the model generated a web entity, which is a short summary of the painting: the piece’s name, the artist’s name, the style and the technique.

To generate the prompts, we used GPT-40 mini model. For each web entity, the model produced a short prompt that describes the painting: the objects, the colors and the general style.

Finally, we used DALL-E3 to generate paintings from the prompts. Aiming for 1000, we eventually managed to generate 898 samples. The reason is that some prompts were denied by DALL-E since they were inappropriate (for example nude art). A demonstration to summarize our pipeline with “Portrait of Hendrikje Stoffels” by Rembrandt:



Our data notations: is a text prompt, is a generated painting, and is the reference (original painting). is the reference dataset, is the test dataset.

##### Dataset Split

The proportion of datasets in splitting was a key hyperparameter in the paper. We could have learned the optimal splitting, but we eventually decided focus on more important tasks. We fixed so that the BH procedure will have a sufficient number of samples to conduct the statistical testing. Left with , we split the train and calibration sets approximately similar to the experiments in the paper:

##### Alignment score function

To compute the alignment score , we had to come up with a way to compare the generated images to their references. We experimented a few similarity measurements to test if the generated image and the reference are visually akin, and evetually decided to measure 4 aspects of similarity:

1. **Structural similarity** – we computed the SSIM index, which aims to approximate an objective image quality based on structures that are perceived by human visual system. It performed poorly on high dimensions, so we first resized the images to a smaller size and transformed them to black & white pixels.
2. **Style similarity** – inspired by the style loss formulation of the Neural Style Transfer task, which measures the correlation between features after each layer, we computed the L1 distance of the gram matrices of the images.
3. **Features similarity** – we used the pre-trained VGG16 deep learning model. We fed the images to the model to extract features and produce image embeddings, which we then computed their cosine similarity.
4. **Color similarity** – we generated a color palette of the 10 most bold colors of each image and measured the palettes’ distance after transforming the RGB colors to the CIELAB color space. It is designed to be perceptually uniform; the Euclidean distance between two colors in this space more accurately reflects perceived differences.

For each aspect of the above, we calculated a score between the generated image and the reference. To determine the quality of the scores, we sampled another 10 independent images and 10 independent random noises and calculated the score between them and the generated image (total of 21 scores). If the generated image and the reference have better scores than the generated image and all the other samples, they are most likely to be aligned. We summed the indices of the former in the sorted scores array over each similarity aspect and normalized it to a value between 0 and 1. Formally:

Where is the array of scores of the aspect , the score between the generated image and the real reference is , and is the index of in the sorted .

Demonstrating again on the “Portrait of Hendrikje Stoffels” by Rembrandt, the SSIM score was 0.01476, the style distance was 2.9908, the features embeddings cosine similarity was 0.6715, and the palette distance was 9.4653. Those are not optimal scores by no means, but they were the highest among all the other false references, hence the alignment score was 1.0. Visually, they indeed seem very similar. To sense the color similarity, we displayed the color palettes below.



##### Features used for learning

As discussed in the previous section, identifying informative features for training the alignment predictor is crucial, since the power of Conformal Alignment depends on how well discerns those against those . Unlike the paper’s research, we could not extract DALL-E’s confidence on his generated outputs. In addition, refining the semantic and lexical properties of text using similarity measurements and LLMs like applied in the paper was not a good choice for our context, since we had to test the alignment of an image.

Our primary concern was whether the generated painting aligns with the original artist's style. To cope, we trained our own artist classifier on Google Colab’s GPUs. We implemented the architecture of ResNet50 neural network and trained it on the reference paintings dataset of the 10 artists we selected for our task. The output of the model’s inference is softmax probabilities, indicating which artist has most likely drawn the input painting. We anticipated that if the generated painting is truly similar to the artist’s art style, our model would be able to classify it correctly.

Our second task was to determine whether the generated painting matches the prompt requirements. To do so, we used Facebook’s pre-trained DETR (DEtection Transformer) model with ResNet-50 backbone to detect the objects in the generated painting. After receiving a list of detected objects (regardless of their positions), we counted how many of these objects the prompt and the image have in common as follows: For each object we checked if the prompt contains it. If not, it still doesn’t mean that they are not aligned; For example, a common scenario was where the object detection model recognized a “person” in the image, while the prompt required a dancer. Even though the prompt doesn’t contain the word “person”, a dancer is semantically contained. To address this problem, we used Facebook’s pre-trained NLI-based zero shot text classification model named BART-Large-MNLI to infer whether the prompt was asking to generate the object.

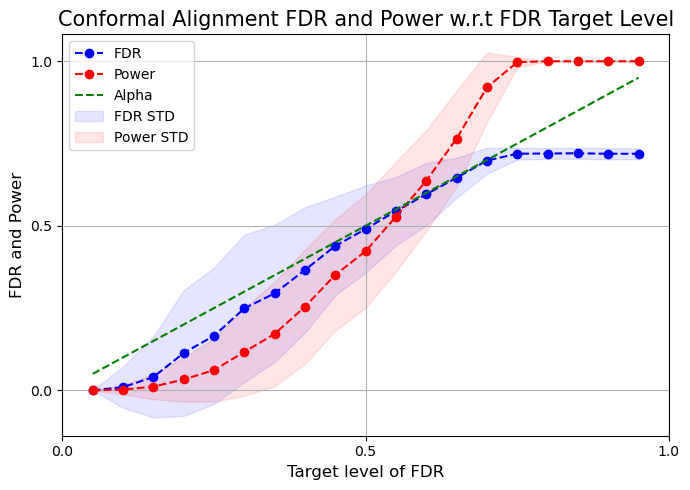
Overall, the feature vector of each generated painting was a concatenation of the artist classifier’s output, a one hot vector indicating the ground truth artist (extracted from the prompt), and the number of shared objects between the prompt and the image.

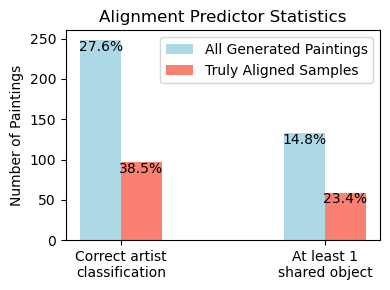
##### Alignment predictor fitting algorithm

To train the alignment predictor , we used XGBoost, one of the most popular machine learning frameworks among data scientists. We ran logistic regression with binary cross entropy loss, since we can consider it as the probability to be aligned and learn its distribution.

# Results

With alignment level of , we conducted 500 independent random data-splits and computed the Power and the FDR over 20 different values of FDR target level:

 Although the results demonstrate a strict control over the FDR level, the standard deviation is relatively large. Like discussed in the paper, that might indicate that we didn’t have a sufficient sample size. The Power we experience is also sub-optimal, compared to the results shown in the paper. As we mentioned in previous section, the features used to train play a vital role in the power of selection. The Power curve indeed reminds the poor features curves in figure 5. Hence, we took a closer look on the ability of the features in measuring alignment.

Both the artist classification performance and the object detection performance were not in our favor: among all the 898 generated paintings, 248 were classified to the correct artist, while among 252 truly aligned samples, only 97 generated paintings were classified correctly. This means that it might mislead the alignment predictor to think that a true classification is reversely correlated to alignment. Furthermore, among all the 898 generated paintings, 133 share at least 1 semantic object with the prompt, whereas among 252 truly aligned samples, only 59 generated paintings hold that property. The reason is that many generated paintings feature abstract elements, lacking distinct lines or clear separation from the background.

However, the most obvious reason for the poor quality of the results is that DALL-E is generally not aligned to artist’s style. It has a unique style of its own ☺ It’s evidential that nearly all the generated paintings had brighter and more vivid colors than their reference. Most of the paintings displayed an impressionist brushstroke technique typical of oil paintings, despite the original reference having a different style. Moreover, for some paintings, the prompt we generated with GPT didn’t describe the painting correctly, which led to imprecise generation. For example:

Van-Gogh’s “Houses seen from the back” (the generated on the right), alignment score 0.4285. Th artist classifier decided it is a painting by Van-Gogh with probability 0.03, no objects detected. The prompt required to feature “charming houses in vibrant colors, contrasting the white snow”. Obviously, that is not a good description of the painting.

Interestingly, 31 aligned samples were classified correctly and shared at least 1 semantic object with their prompt, 26 of them were paintings by Rembrandt, 4 Titian’s and 1 Paul Gauguin’s. This is not a surprise; Rembrandt’s paintings are easy to learn, since they are more realistic (their objects are likely detectable) and their colors are uniformly in shades of brown, distinguishable from the others.

# Conclusion and Future Works

# References

Kaggle dataset: <https://www.kaggle.com/code/paultimothymooney/collections-of-paintings-from-50-artists/input?select=images>

Artist classifier: <https://medium.com/analytics-vidhya/predict-artist-from-art-using-deep-learning-9f465f8879d7>

SSIM: Zhou Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," in IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, April 2004, doi: 10.1109/TIP.2003.819861.

Features similarity: <https://medium.com/@developerRegmi/image-similarity-comparison-using-vgg16-deep-learning-model-a663a411cd24>

Style similarity: [Li, Y., Wang, N., Liu, J., Hou, X., Demystifying neural style transfer, in: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, International Joint Conferences on Artificial Intelligence Organization. p. 2230–2236. 1](https://arxiv.org/abs/1701.01036)

CIELAB color space: <https://en.wikipedia.org/wiki/CIELAB_color_space>

Zero shot model: <https://huggingface.co/facebook/bart-large-mnli>

Object detection model: <https://huggingface.co/facebook/detr-resnet-50>