ざっくり各ファイルの説明

Preprocessing -> データの事前処理　Dataset.zipを投げるとnpy(numpyファイル)で返ってくる。このとき、ShapEのschedulerまでは通ってる。

ForRe\_2 -> 訓練　Prepro(ryで投げたDataset.zipと、返ってきたoutput\_mesh.npyを投げる。それ以外はコメントを参照のこと

ForRe\_3 ->訓練結果のモデルのパラメータから推論を”したかった”　現状、追加したパラメータがうまく機能しないため改善の必要あり。

Dataset.zip ->訓練に使ったデータ　ShapeNetの02691156.zipから、screenshotが含まれている、かつモデルの頂点数が閾値以上のものを選定し、そのうち100個をzipにまとめてる。

Preprocessingに選定を行うコードがあるから、その結果をDLすれば他のデータ(ShapeNetなら)でもok

実験結果　->実験結果。　たまに詳しいデータが消えているが、おそらくファイル直下のシートに書かれたデータのどれか(no?のように分けておくことを推奨)

研究で使ったサイトのURL達(すべてが有用というわけではないし、ブックマークしていたものなのでこれ以外にも参考にしたサイトはおそらくある)

[HuggingFace-Shap-E](https://huggingface.co/openai/shap-e)

[GitHub - openai/shap-e: Generate 3D objects conditioned on text or images](https://github.com/openai/shap-e)

[Lora](https://arxiv.org/abs/2106.09685)

[GitHub - microsoft/LoRA: Code for loralib, an implementation of "LoRA: Low-Rank Adaptation of Large Language Models"](https://github.com/microsoft/LoRA)

[GitHub - cloneofsimo/lora: Using Low-rank adaptation to quickly fine-tune diffusion models.](https://github.com/cloneofsimo/lora)[初めてでもできる！ 自宅PCのGPUを「Google Colab」から使い、生成AIをカンタンに動かす方法【イニシャルB】 - INTERNET Watch](https://internet.watch.impress.co.jp/docs/column/shimizu/1531958.html)

[Perceiver解説 | リーディング・エッジ社　旧・研究開発部ブログ](https://leadinge.co.jp/rd/2022/08/08/2779/)

[GitHub - lucidrains/perceiver-pytorch: Implementation of Perceiver, General Perception with Iterative Attention, in Pytorch](https://github.com/lucidrains/perceiver-pytorch/tree/main)

[GitHub - yardenfren1996/B-LoRA: Implicit Style-Content Separation using B-LoRA](https://github.com/yardenfren1996/B-LoRA)

[LLMチューニング手法「LoRA」のポイントと活用例 #AI - Qiita](https://qiita.com/DeepMata/items/cb4ff18c1e0548bdb844)

[GitHub - huggingface/peft: 🤗 PEFT: State-of-the-art Parameter-Efficient Fine-Tuning.](https://github.com/huggingface/peft)

[OpenCALM-7B を PEFT を使って LoRA でファインチューンする.ipynb · GitHub](https://gist.github.com/niw/b8e8509147e27b943b4d9af01bea91cf)

[ShapeNet/ShapeNetCore · Datasets at Hugging Face](https://huggingface.co/datasets/ShapeNet/ShapeNetCore)

[日用品データセット + 3Dデータベース​ | 人工知能研究センター](https://www.airc.aist.go.jp/achievements/ja/p-007.html)

[README.md · VAST-AI/LD-T3D at main](https://huggingface.co/datasets/VAST-AI/LD-T3D/blob/main/README.md)

[NeRF検討用データセットをBlender Pythonスクリプトで作成する方法 #DeepLearning - Qiita](https://qiita.com/live2d_dev/items/7e1628670f299036cf5f)

[diffusers/examples/text\_to\_image/train\_text\_to\_image\_lora.py at main · huggingface/diffusers · GitHub](https://github.com/huggingface/diffusers/blob/main/examples/text_to_image/train_text_to_image_lora.py)

[Trainerについてのメモ - Planaria Work Log](https://planaria.page/blog/?p=1090)

[peft/examples/conditional\_generation/peft\_lora\_seq2seq.ipynb at main · huggingface/peft · GitHub](https://github.com/huggingface/peft/blob/main/examples/conditional_generation/peft_lora_seq2seq.ipynb)

[3Dモデルも生成AIで作れる時代！おすすめAIツールの料金や特徴まで徹底解説 | WEEL](https://weel.co.jp/media/3d-model-generation-ai/#index_id8)

[Mialiu/nerf · Datasets at Hugging Face](https://huggingface.co/datasets/Mialiu/nerf)

[fpsample · PyPI](https://pypi.org/project/fpsample/)

[Farthest Point SamplingCCT-recruit | CCT-recruit](https://recruit.cct-inc.co.jp/tecblog/point-cloud-processing/farthest-point-sampling/)

[diffusers/src/diffusers/pipelines/shap\_e/pipeline\_shap\_e.py at main · huggingface/diffusers · GitHub](https://github.com/huggingface/diffusers/blob/main/src/diffusers/pipelines/shap_e/pipeline_shap_e.py)