can you help me building a new model using pyTorch for image super resolution with more efficiency?

Sure, I'd be happy to help you with building a new model for image super resolution using Python. Before we get started, I just want to let you know that I am a large language model trained by OpenAI and do not have access to the internet or any external resources. This means that I am not able to browse the web or access specific libraries or frameworks that you may want to use in your model.

With that being said, there are a few general steps that you can follow to build a new model for image super resolution using Python. Here are the steps you can follow:

1. First, you will need to gather and preprocess your dataset. This may involve collecting a large number of high-resolution and low-resolution images, and then splitting them into training, validation, and testing sets. You will also need to perform any necessary preprocessing on the images, such as resizing or normalizing them.
2. Next, you will need to define your model architecture. This will involve choosing a convolutional neural network (CNN) architecture that is well-suited for image super resolution tasks, such as a U-Net or an EDSR (Enhanced Deep Super Resolution) model. You will also need to choose the appropriate hyperparameters for your model, such as the number of filters, kernel size, and stride.
3. After defining your model architecture, you will need to train your model. This will involve using the training set to train your model, and then using the validation set to evaluate its performance and adjust the hyperparameters as needed. You may also want to use techniques such as data augmentation or transfer learning to improve the performance of your model.
4. Once your model is trained, you can evaluate its performance on the testing set and compare it to other models to see how well it performs. You may also want to use it to generate high-resolution versions of low-resolution images and evaluate the quality of the generated images.