CAD Operation Sequence

67284 Project Machine Learning and Deep Learning
Summer Semester 2022

Saurabh Varshneya Vedaant Joshi

October 25, 2022

1 Summary

The main idea of the Project was to train a network on the basis of the image features (CAD 3d image) and the sequence of the operations available (text data). The network after training would take image features as input and predict the CAD operation sequence.

Two approaches were used for the above-mentioned tasks :

- Training the model with a 3D CNN - LSTM architecture.
- Training the model with a 3D CNN - Decoder Transformer architecture.

In both of the above approaches, the same 3D-CNN architecture is used.

2 Paper

2.1 CSGNet : Neural Shape Parser for Constructive Solid Geometry

- The main idea of this paper is to combine supervised pretraining, reinforcement learning, reward design, and post-optimization of modeling parameters to obtain a generalized model.
- The neural network is trained on the basis of a large synthetic data set which has been automatically generated by the CSG programs.
- The synthetic data set is created by sampling random programs containing different number of primitives (shapes).
- The instructions to generate an 3D object, follows a standard post-fix notation. Hence, all the sequences predicted should follow this notation. Which indeed also implies that the order of instructions matters a lot. If the sequence in the post-fix notation changes, the final object that is generated (using that sequence) will be completely different. Note: It is quite possible that changing the order might give the same output, but that's not always true.
- The data is stored as post-fix notation [cy(48,48,32,8,12)cu(24,24,40,28)+].
- In the 3D synthetic dataset, there are data of different program lengths. For,
  - Program length = 3, 100k Train, 10k Val and 20k test dataset
  - Program length = 5, 200k Train, 20k Val and 40k test dataset
  - Program length = 7, 400k Train, 40k Val and 80k test dataset
- Metrics used for the evaluation of the trained models are Chamfer Distance, IoU (Intersection over Union) and also in the form of rewards.
  - The Visual Similarity between two shapes (target and predicted) is measured using the Chamfer Distance between points on the edges of each shape.
  - The rewards be primarily designed to encourage the visual similarity of the generated program with the target. (Used with RL)

  - While testing the model, IoU metric is used between the target and the predicted sequence.

- Chamfer Distance (CD)
  - This metrics is used to measure the visual similarity between two shapes.
  - The CD between two point sets, X and Y, is defined as follows:

$$Ch(x,y) = \frac{1}{2|X|}\sum_{x\in X}\min_{y\in Y}||x-y|| + \frac{1}{2|Y|}\sum_{y\in Y}\min_{x\in X}||x-y||$$

  - $Ch(x,y) \in [0,1]$
  - Lower the value of CD better is the model.
  - Chamfer distance is used between two point clouds.

- The 3D data is available in the following format :
  - sp(x,y,z,r), cu(x,y,z,r) and cy(x,y,z,r,h). These are the three primitives present in the dataset.
  - Here, (x,y,z) are the center (location) of the shapes in the 3D space.
  - r denotes the radius for sphere and cylinder whereas r specifies size for the cube.
  - h denotes the height of the cylinder.

3 Basic CNN

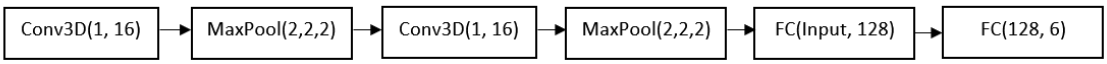
3.1 Classes

Basic classifier model for classifying sequence of one operation into following 6 labels :

- (SP,SP)
- (SP,CU)
- (CU,CU)
- (CY,CY)
- (CY,SP)
- (CY,CU)

The loss function used during all the training iteration is CrossEntropyLoss.

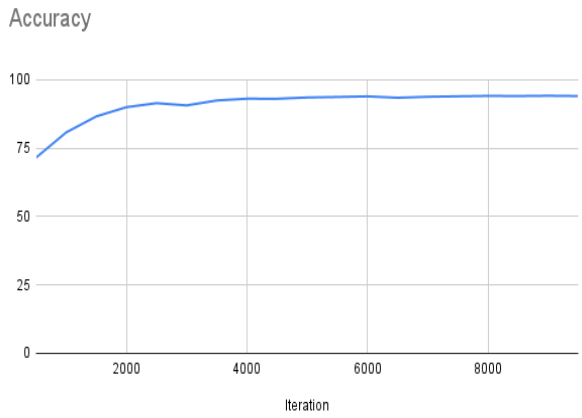
3.2 Architecture

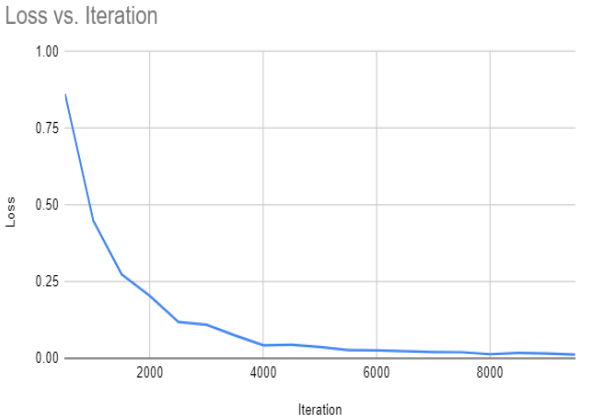


3.3 Data Points

Trained with 20k Voxel data points.

3.4 Results



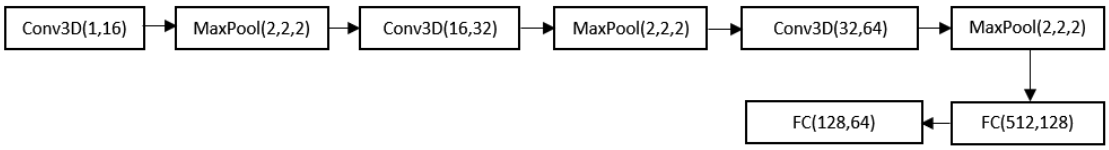


4 CNN-LSTM Model

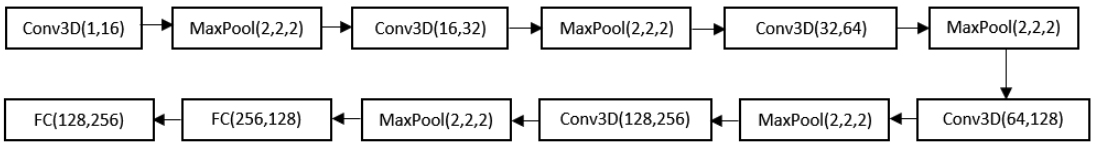
- Prediction of sequence of primitives [only the sequences and not the parameters (x, y, z, r, h)].
- Two different models being used for training on this data. Basic CNN-LSTM model and CNN-LSTM model with more layers.

4.1 Architecture

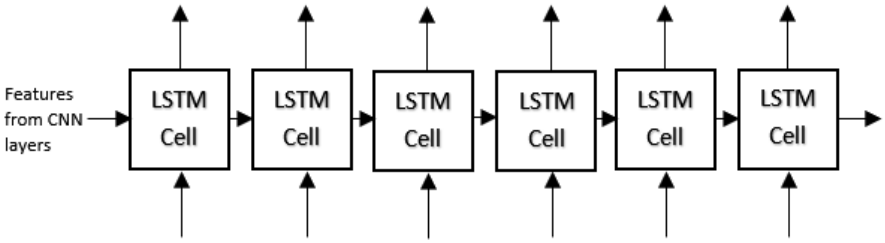
4.1.1 Model 1 (CNN)



4.1.2 Model 2 (CNN)



4.1.3 LSTM Architecture



4.2 Results

CNN_LSTM Model	#epochs	Number of data points (80:10:10)	BLEU SCORE n_gram = 2	BLEU SCORE n_gram = 3	Loss
model_1	100	1K	0.39842314	0.26031667	0.9253711037
		10K	0.57313013	0.50983083	0.5252447592
	200	1K	0.32747775	0.31770915	0.7769873135
model_2	100	1K	0.48278555	0.3100113	0.8284203723
		10K	0.80891496	0.7173402	0.3000257445
	200	1K	0.496473	0.39392886	0.6936606131

5 Vorplan Dataset

The dataset is similar to the first dataset. 3D Image features and Sequences of instruction are the inputs to the network.

Two different approaches are used for training Vorplan Dataset:

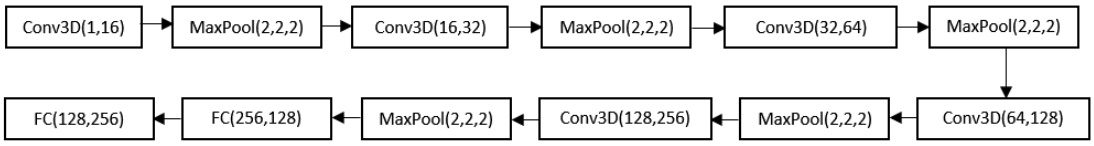
- CNN-LSTM architecture
- CNN-Decoder Transformer architecture

Training is done on two different datasets:

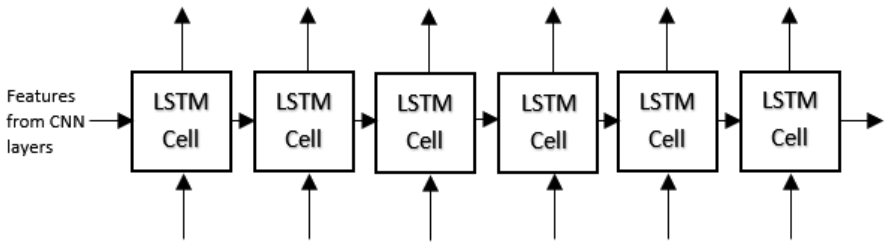
- Dataset 1 consists of 166 data points,
- Dataset 2 consists of 254 data points.

5.1 Architecture

5.1.1 CNN Architecture

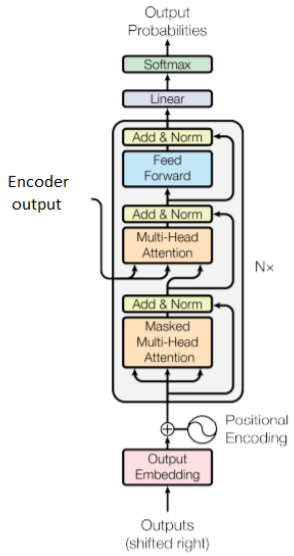


5.1.2 LSTM



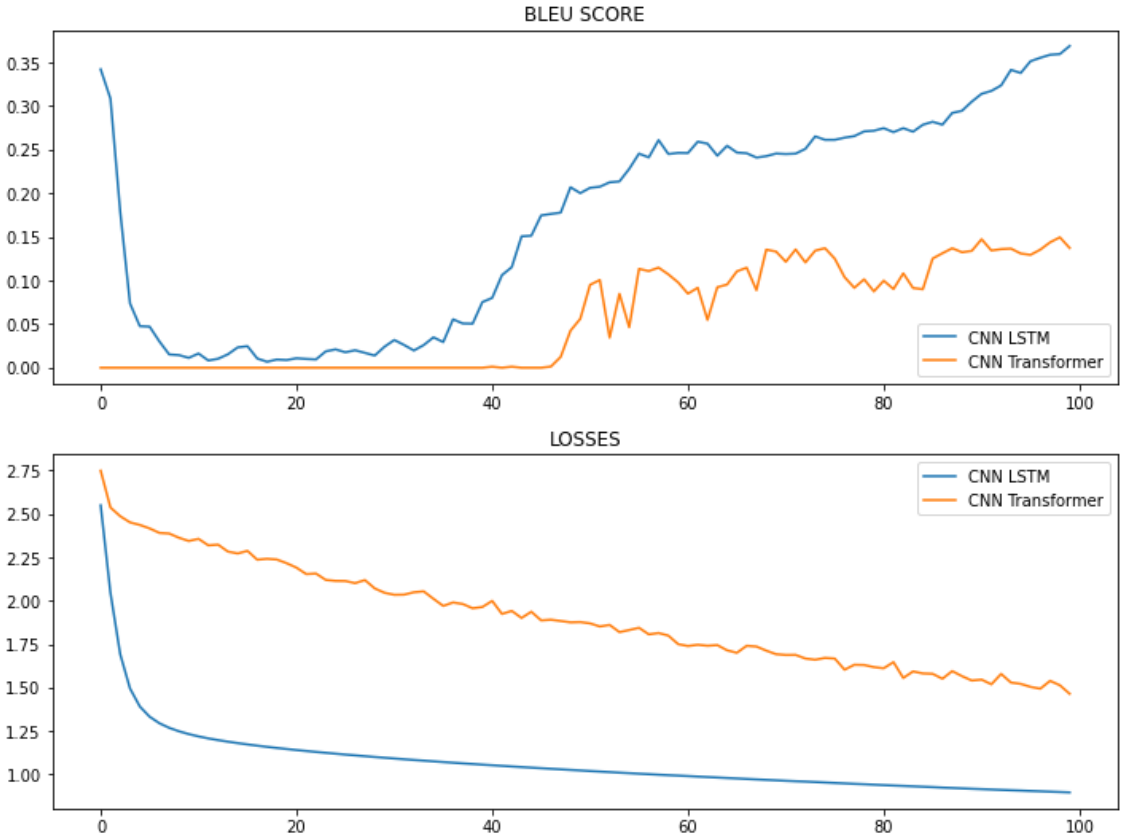
5.1.3 Transformer Decoder

Transformer Model referenced from following GitHub Repository - senadkurtisi/pytorch-image-captioning



5.2 Results

- CNN-LSTM and CNN-Transformer on Dataset 1 with BLEU Score calculated using n-gram=3



- CNN-LSTM and CNN-Transformer on Dataset 2 with BLEU Score calculated using n-gram = 1, 2, 3, 4