Img2Mol - Converts molecule images to SMILES

Dosapati Sri Anvith DOSAPATI.SRI@STUDENTS.IIIT.AC.IN
Student ECE
International Institute of Information Technology
Hyderabad, India

Avyukta Manjunatha Vummintala AVYUKTA.V@RESEARCH.IIIT.AC.IN
Student ECD
International Institute of Information Technology
Hyderabad, India

Lokesh Venkatachalam LOKESH.V@RESEARCH.IIIT.AC.IN
Student CSD
International Institute of Information Technology
Hyderabad, India

Mugundan Kottur Suresh MUGUNDAN.KOTTUR@STUDENT.IIIT.AC.IN
Student CSE
International Institute of Information Technology
Hyderabad, India

Tutors: Anir Vinaya G.

Abstract

This report deals with the automatic recognition of molecules from its graphical depiction. The rapidly expanding field of computational chemistry and drug discovery has made the problem significant and challenging. In this report, our main aim is to study the reference paper and implement the model. Additionally, we have also developed a front-end for the same.

Keywords: CDDD, SMILES, Convolutional Neural Networks

Introduction

The 20th century witnessed the rise of computational technology and its sister sciences. The tremendous increases in computing power have also influenced other research fields like biology, physics, chemistry, etc., creating diverse fields like computational biology, computational physics, and computational chemistry. In our report, we deal exclusively with a problem of computational chemistry. More specifically, we aim to create a machine-learning model that can recognize a molecule from its image. Our reference paper [1] has been published in the Chem Sci. journal 2021 and provides a state-of-the-art solution for the abovementioned problem. The crux of the answer is a clever and modular implementation of convolutional neural networks.

0.1 Motivation

Papers often store important molecules as images rather than sequences for immediate use by other papers or research. Therefore, understanding the paper and decoding images in research papers leads to a waste of time. This model helps automate that process.

Theory

0.2 Basic Notions

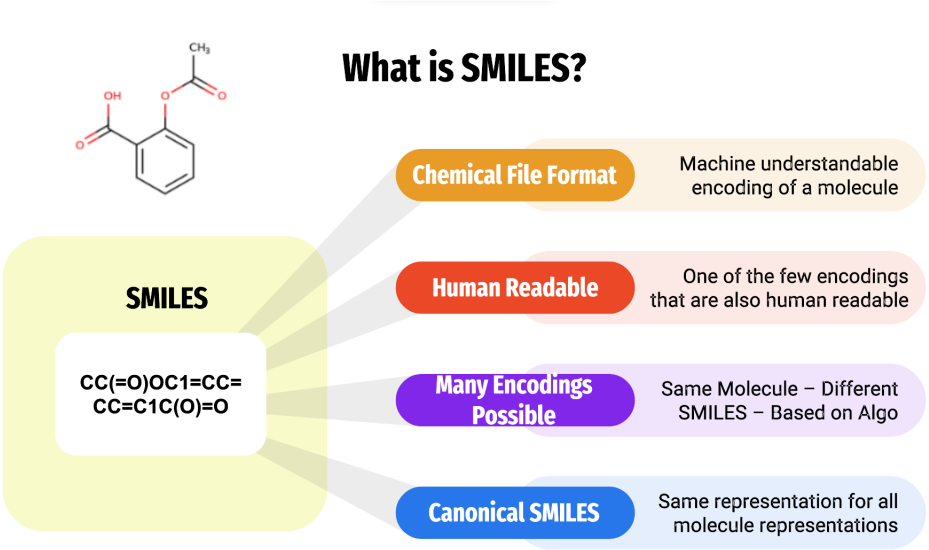


Figure 1: Description of various chemical representations

0.3 Previous Work

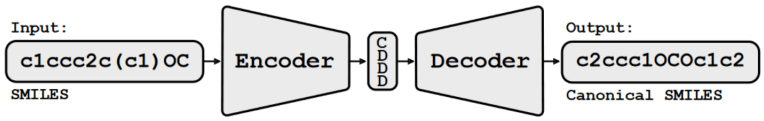


Fig. 2 Conceptual view of the *CDDD* autoencoder. An input SMILES string representing a molecule is encoded into a 512-dimensional feature vector (the *CDDD* embedding). The decoder was trained to produce canonical SMILES from the *CDDD* embedding.

The authors reuse a model from their previous paper in this paper and do not dwell deep into the exact nature of the model. The last research paper converts any SMILES representation to a canonical SMILES representation. It did so using an auto encoder that

had two parts. An encoder that converts SMILES to an intermediate CDDD embedding and a decoder that converts the CDDD embedding to canonical SMILES.

0.4 State-of-the-art-model

Our reference paper creates a state-of-the-art model based on convolutional neural networks. The entire model is modular in design and has the following components:

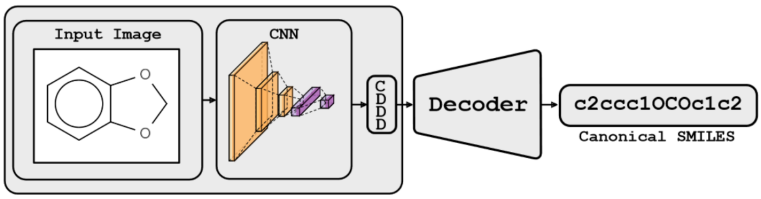


Fig. 3 Overview of the *Img2Mol* workflow for molecular optical recognition. The left part is what is newly trained in this work, while the pretrained *CDDD* decoder is used to obtain canonical SMILES.

1. Input Image: The image of a molecule with certain limits upon size and number of atoms present.
2. CNN: The convolutional neural network is the central part of the model, as all the learning happens in this stage.
3. CDDD: This is a powerful continuous molecular descriptor.

The current Img2Mol model reuses the previous paper's decoder and trains a new CNN model that converts images to CDDD embedding. Let us now focus on CNN:

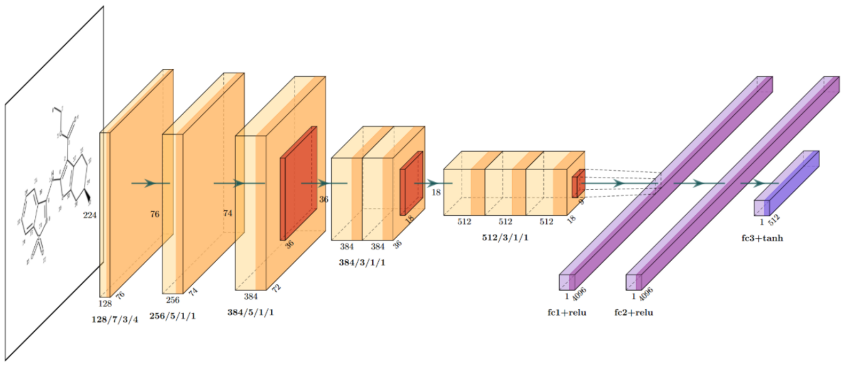
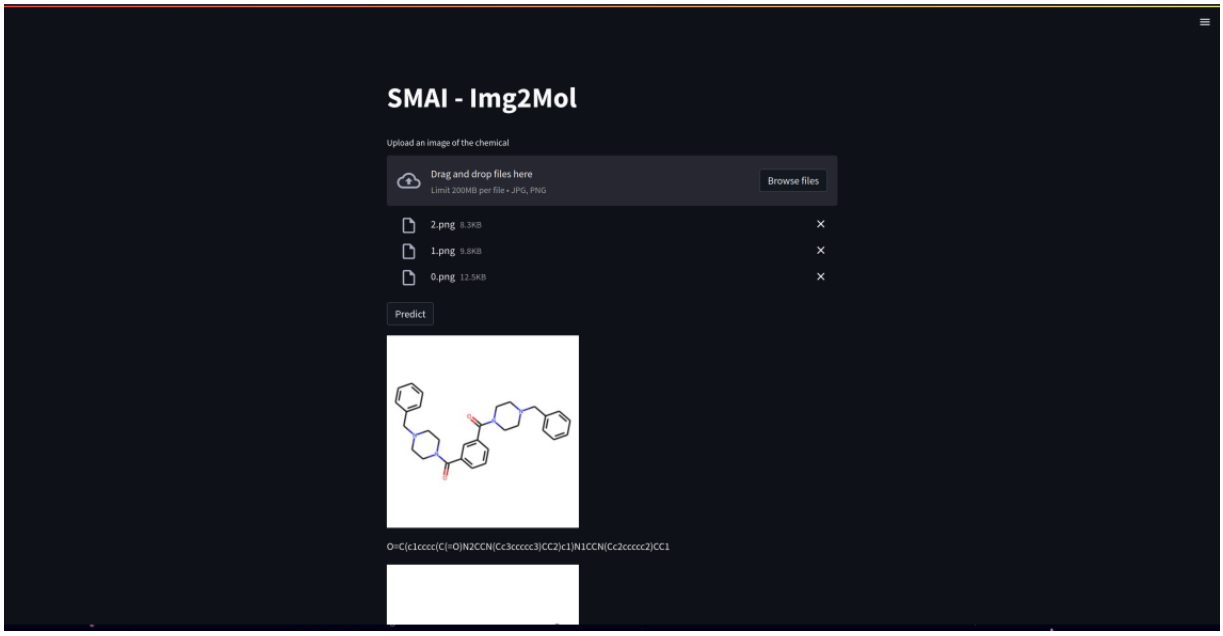


Figure 2: Layers of the Convolutional Neural Network

As depicted in the figure, the convolutional neural network is structured into 11 layers, each having its structure and activation function. A detailed description of each layer is as follows:

- Layer 1: This is convolutional layer of dimensions 128/7/3/4.
- Layer 2: This is convolutional layer of dimensions 256/5/1/1.
- Layer 3: This is a convolutional pooling layer of dimensions 384/5/1/1.
- Layer 4: This is a convolutional layer of dimensions 384/3/1/1.
- Layer 5: This is a pooling layer of dimensions 384/3/1/1.
- Layer 6: This is a pooling layer of dimensions 512/3/1/1.
- Layer 7: This is a pooling layer of dimensions 512/3/1/1.
- Layer 8: This is a pooling layer of dimensions 512/3/1/1.
- Layer 9: This is a fully connected layer of dimensions 4096 by 1 and ReLU activation function.
- Layer 10: This is a fully connected layer of dimensions 4096 by 1 and ReLU activation function.
- Layer 11: This is a fully connected layer of dimensions 512 by 1 and tanh activation function.

Front End



We have also developed a front end for the model. The user can use the front end in the following manner:

1. Upload the image to be recognized.
2. Click on the 'Predict' button and wait for the processing.
3. View the list of possible outputs.

Training and Testing

We used the Img2Mol dataset for training and testing. The model is trained on 60,000 images from the abovementioned dataset. We used the NVIDIA VCLE A100 16 GB VRAM. The following table summarizes the training, testing, and validation data split:

Train+Validation : Test	Train : Validation	Train : Validation : Test
80:20	90:20	72:8:20

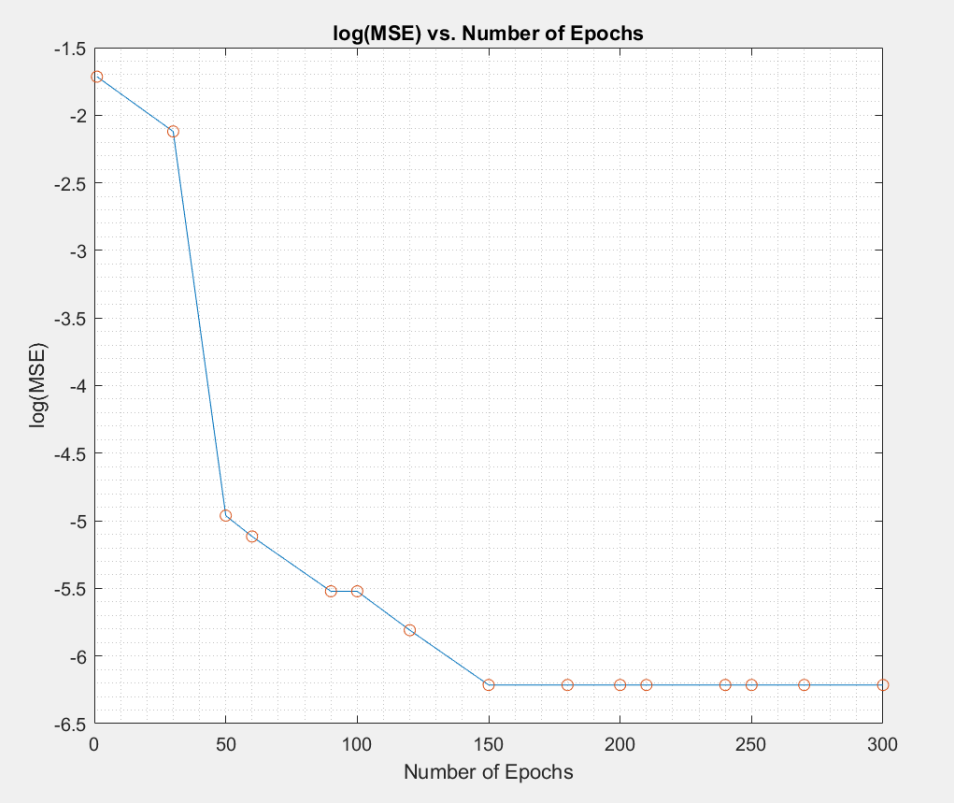
In summary:

- Train: 43200 (38 batches)
- Validation: 4800
- Test: 12000

We ran the model for 300 epochs. The code is available on our GitHub repo.

Results and Analysis

The training ended with a final loss of 0.002 MSE. This translates to **55**% **accuracy**. The following plot summarizes MSE loss vs. the epochs trained



The above plot reinforces the standard heuristic that most of the learning happens in the first 50 epochs. This is illustrated by the sharp decline in MSE in the [0,50] range. We further see that after 150 epochs, there doesn't seem to be any significant decrease in MSE. So, we can possibly infer that that region marks the beginning of overfitting.

For 50 epochs, on the Img2Mol Benchmark dataset, we get 15/5000 exact matches. For 300 epochs, we get 39/5000 exact matches.

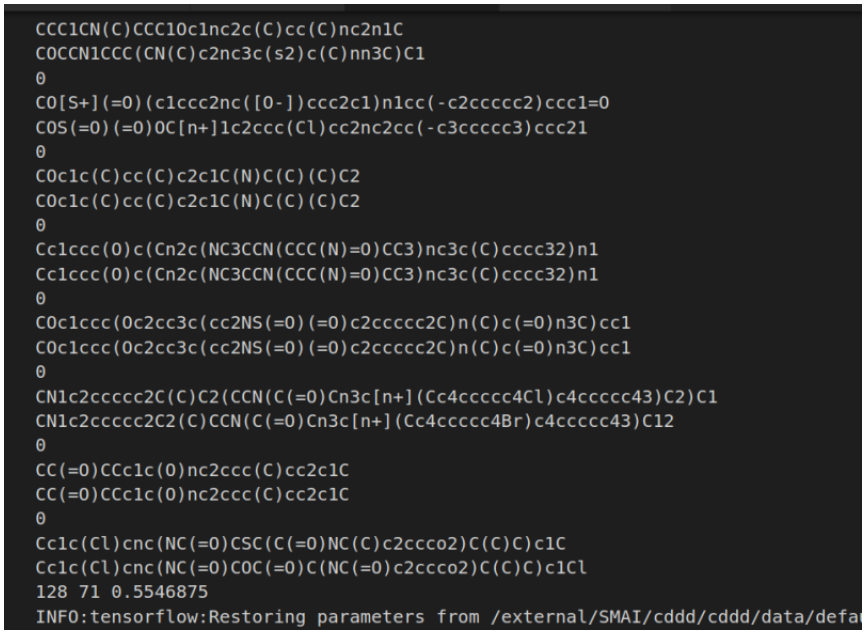


Figure 3: Comparison for test data

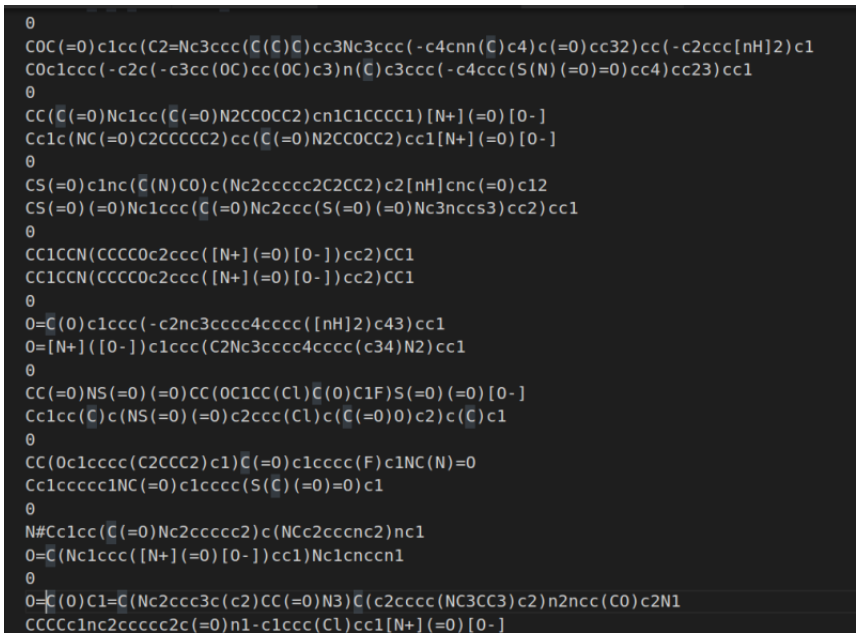


Figure 4: Img2Mol Benchmark Data

Conclusions

In this report, we present our version of Img2Mol. The model learns from several pictorial representations of molecules and aims to predict the CDDD embedding of the depicted

molecules accurately. The model outperforms other baseline approaches in most situations and is much faster (provided GPU hardware.)

Acknowledgements

We thank professor Anoop Namboodiri for allowing us to work on this project. We also thank our Teaching Assitant, Anir Vinaya Gururanjan, for his guidance. Finally, we thank our teammates for their comradeship and good humor, even during the most stressful times.

References

1. *Img2Mol–accurate SMILES recognition from molecular graphical depictions* - Djork-Arne Clevert,Tuan Le, Robin Winter and Floriane Montanari, Chem. Sci., 2021, 12, 14174
2. Our Code:here
3. CDDD Code: here
4. Trained Model

Distribution of Work

Mugundan Kottur Suresh - Partial Frontend and Backend
Lokesh Venkatachalam - Partial Backend and hosting CDDD server
Dosapati Sri Anvith - Partial Frontend and Backend
Avyukta Manjunatha Vummintala - Reports and presentations