ITSP Final Documentation

Image to LaTeX Converter

Hawkeye

ITS20057

Keywords

CNN, seq2seq, Encoder-Decoder, RNN, Im2latex, PyTorch, LaTeX, Django, REST API, OpenCV

Team Member Name	Roll Number	Email-Id
Sai Saketika Chekuri	190070054	saketikachekuri@gmail.com
Sahasra Ranjan	190050102	sahasraranjan@gmail.com
Prakhar Mittal	190070046	190070046@iitb.ac.in
Sakshi Toshniwal	190070055	sakshiudr2001@gmail.com

Inspiration for Idea

There are many OCR tools available to recognize digits or characters, but most of these can't recognize the various mathematical symbols or equations. We wish to focus on this problem and come up with an effective tool that can not only recognize them but also go the extra mile to convert them into a standard LaTeX form.

Problem Statement

A Machine Learning model for generating LaTeX expression for a given math expression (image) and, a website for the same which can take the image as an input and return its equivalent LaTeX expression.

Existing solutions in the Market

There are a few apps and websites(Mathpix) that can do this task, and a few research papers as well. We've based our idea on one such research paper from Stanford's CS231n course, the link to which can be found here.

Proposed Solution

We decided to build a website where one could upload an image of a math equation, and we shall provide said a person with a complete LaTeX code for cloning the desired equation for his/her research work/homework/ wherever the user wishes to implement, hence making it a lot easier for them to showcase their work digitally.

Brief Description

We have used a CNN encoder RNN decoder hybrid model for reading the formulae from the image input, encoding the images, decoding into appropriate tags, and parsing it into compilable LaTeX code.

Our website takes the input image from the user, and initially preprocesses and resizes it so that it has 3 channels (like the images we trained on). The parameters we've obtained by training our Neural Network are then used to predict the math equation in the preprocessed image, and convert it to its corresponding LaTeX expression.

We also provide a compiled pdf file for showing how the predicted code would look after compiling with a TeX editor.

Progress

Work distribution:

1. Preprocessing was majorly handled by Sakshi
2. Model and training were done by Saketika
3. Sahasra worked on the model and Django REST API
4. Prakhar worked on the front end of the website

Work-flow:

1. For the first phase, we covered the concepts. We decided to use CNN for our model but after suggestions provided by the panel and our Mentor, we shifted to RNN based Seq2seq (Encoder-Decoder) model.

2. For the second phase, since the Seq2seq model was new for us, we covered the concepts in the initial days. We then started working on the model (Preprocessing, Encoding, Decoding and Evaluating). Meanwhile, we build a Django, REST API based website for the same.

3. In the 3rd phase, we've focused on training our models with different hyperparameters, since training each took up around 2 days. We've also completed the front-end of our website and resolved errors we were getting due to input images with a different number of channels and size.

Challenges faced:

1. We've initially started training on Colab, but our training data was too large, so switched to Azure.

2. We were not familiar with the Seq2seq model, it took us a lot of time to figure out what is going on with the model.

3. Training each model for 20 epochs took a very long time-- around 30 hours each.

4. Initially, our website was throwing out errors for user-uploaded images since we've trained only on images of particular sizes and with 3 channels only. This problem was later fixed by us by writing a code to preprocess and resize the image before passing it into the model.

5. Our project still lacks efficiency with general images, we will consider this as our future plan.

Results

Images: SnapMath - GitHub

Video: Here is a short video of our website, which shows in working how an input image is typeset into LaTeX code.

GitHub Repositories: Hawkeye - ITSP - GitHub

Presentation: Final presentation - Hawkeye

Learning Value

This has been a great experience for us, especially since it's our first actual project that involved a huge amount of data. We've learned how to use RNNs, CNNs, and how to implement a seq2seq model using Pytorch. Building our website also gave us an intro to Django and Bootstrap.

Software/ Hardware used

- Google Colab
- Microsoft Azure
- Django
- Pytorch
- Various OpenCV libraries used for Preprocessing
- Github
- VS Code
- Sublime Text

Suggestions for others

- Spend more time reading about the model
- If your model has low accuracy, focus on preprocessing first before modifying the model.

Contribution by each Team Member

Saketika:

- Worked on the model and trained different models on Azure
- Evaluated the 3 different models to see which has the best results

Sahasra:

- Worked on the model
- Created a Django REST API based webpage for LaTeX prediction

Prakhar:

- Worked on the model
- Suggested some model ideas for improving accuracy
- Prepared front-end for the website
- Assisted in debugging for preprocessing

Sakshi:

- Worked on the model
- Created functions to preprocess the input image

References and Citations

- Guillaume Genthial Blog on seq2seq for LaTeX generation
- PyTorch implementation for the model described in Guillaume's blog (author - Luopeixiang)
- Gidi Shperber, (2018). A gentle introduction to OCR - Towards Data Science
  Learned the underlying principles for an OCR.
- R Sanjeev Kunte, (2008). OCR - Kannada Script: Theory behind segmentation for OCR

PyTorch:

- Official PyTorch Tutorials
- PyTorch for Deep Learning and Computer Vision: Learnt how to implement various ML algorithms using PyTorch

Django:

- Django - Documentation
- Django REST framework - Tutorials

Front-end (Bootstrap + CSS):

- w3schools

Disclaimer

Fair use of:-

- Harvard- im2latex 100k dataset
- Guillaume Genthial Blog on seq2seq for LaTeX generation
- Implementation for the model described in Guillaume's blog

Licenses

- Luopeixiang-im2latex
- Microsoft Azure
- Django REST
- Bootstrap templates