CS 334: Machine Learning
Emory University
Fall 2023
Aadi Waghray and Joshua Sun

Project Proposal: Converting Handwritten Math to LaTeX, 2023 Edition

Aadi Waghray and Joshua Sun

Introduction

Using machine learning to parse an image to text is an extremely common problem. For example, the MNIST dataset is a common introduction to neural networks. [2] The dataset was released to parse the numerical values written on checks. Google's use of reCAPTCHA also demonstrates how optical character recognition worth pursuing. [1] For mathematical academics, it would be extremely useful to convert handwriting for mathematical equations, as much of math is still done by hand to this day.

Dataset

In our project, we aim to address the open problem of optical character recognition. More specfically, we want to address the open problem of converting handwritten mathematical expressions into LaTeX. To do so, we use ICDAR's 2023 Competition on Recognition of Handwritten Mathematical Expressions (CHROME) dataset. This dataset is partitioned into three tasks: online recognition, offline recognition, and bimodal recognition. Online recognition attempts to determine the characters as they are being written. For this purpose, the partitioned dataset for this task includes inkml files, a file format that stores the vectorized data of each expression written by hand, from previous iteratoins of CHROME.[6] In our case, we will likely use the data set aside for task two, offline recognition. This partition of the dataset incudes rendered inkml files and scanned immages. It is meant for models that give the LaTeX expression for a written expression based on the entire expression rather than individual strokes. From the above explanation, it should be obvious bimodal deals with both. Also, it should be clear we are choosing offline recognition given the apparent complexity of online recognition.

While the general problem seems simple, it has several components. A given model has to identify the strokes of a given character, the positioning of the character, and the relationship between the characters to be able to find a suitable LaTeX expression. Based on the work of Sakshi and Kukreja, there are five

main areas of challenges with current approaches: Preprocessing, Input/Output formats and representations, Recongnition model challenges, Compartive analysis and performance evaluation, and Hardware challenges. [3] Based on the content of the class, we can ignore challenges pertaining to hardware, and Input/ Output formats and representation. Between the final three areas, we are narrowing down the exact question we would like to answer. Currently, we believe answering questions relating to the Comparative analysis such as developing guidelines for metric standardization and "forgiven"symbols based on similiarity. Alternatively, we could deal with recognition model challenges by comparing different neural network structures or determining and comparing optimal hyperparameters between models. Finally, we could consider the effect of multiple preprocessing techniques such as illumination, curve and blur, low resolution, noice, and occlusion on model accuracy. For our final project, we choose the final category, considering the effect of preprocessing techniques on model accuracy.

Model

Having narrowed down the problem to model accuracy based on preprocessing technique, we will consider the types of neural networks we will use. Given the informal nature of this assignment, we will take a random group of networks that are used in field. First, we will use WAP (Watch, attend, and parse). It was created in from CROHME 2016 with a few papers written about it.[8] [7] Second, we use another DenseNet based convolutional neural network (CNN) implemented in a paper by Wang and Shan. [5] At the moment, we are still looking into alternate models to use in our project and will finalize the models used in our project following a more thorough literature review.

References

[1] Luis von Ahn et al. "reCAPTCHA: Human-Based Character Recognition via Web Security Measures". In: *Science* 321.5895 (2008), pp. 1465–1468. DOI: `10.1126/science.1160379`. eprint: `https://www.science.org/doi/pdf/10.1126/science.1160379`. URL: `https://www.science.org/doi/abs/10.1126/science.1160379`.

[2] Yann LeCun, Corinna Cortes, and CJ Burges. "MNIST handwritten digit database". In: *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist* 2 (2010).

[3] Sakshi and Vinay Kukreja. "A dive in white and grey shades of ML and non-ML literature: a multivocal analysis of mathematical expressions". In: *Artificial Intelligence Review* 56.7 (July 2023), pp. 7047–7135. ISSN: 1573-7462. DOI: `10.1007/s10462-022-10330-1`. URL: `https://doi.org/10.1007/s10462-022-10330-1`.

[4] Amit Schechter, Norah Borus, and William Bakst. *Converting Handwritten Mathematical Expressions into LATEX*. 2017. URL: `https://cs229.stanford.edu/proj2017/final-reports/5241761.pdf`.

[5] Hongyu Wang and Guangcun Shan. *Recognizing Handwritten Mathematical Expressions as LaTex Sequences Using a Multiscale Robust Neural Network*. 2020. arXiv: `2003.00817`.

[6] Yejing Xie et al. *ICDAR 2023 CROHME: Competition on Recognition of Handwritten Mathematical Expressions*. 2023. URL: `https://crohme2023.ltu-ai.dev`.

[7] Jianshu Zhang, Jun Du, and Lirong Dai. "Multi-scale attention with dense encoder for handwritten mathematical expression recognition". In: *International Conference on Pattern Recognition*. 2018, pp. 2245–2250.

[8] Jianshu Zhang et al. "Watch, attend and parse: An end-to-end neural network based approach to handwritten mathematical expression recognition". In: *Pattern Recognition* 71 (2017), pp. 196–206.