IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 21 March 2024, accepted 16 May 2024, date of publication 23 May 2024, date of current version 6 June 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3404834

RESEARCH ARTICLE

MathNet: A Data-Centric Approach for Printed Mathematical Expression Recognition

FELIX M. SCHMITT-KOOPMANN[1,2], ELAINE M. HUANG[2], HANS-PETER HUTTER[1], (Member, IEEE), THILO STADELMANN[3,4], (Senior Member, IEEE), AND ALIREZA DARVISHY[1]

[1]Institute of Computer Science, Zurich University of Applied Sciences (ZHAW), 8401 Winterthur, Switzerland
[2]People and Computing Laboratory, University of Zürich, 8050 Zürich, Switzerland
[3]Centre for Artificial Intelligence, Zurich University of Applied Sciences (ZHAW), 8400 Winterthur, Switzerland
[4]European Centre for Living Technology (ECLT), 30123 Venice, Italy

Corresponding author: Felix M. Schmitt-Koopmann (scmx@zhaw.ch)

This work was supported by the Bridge Discovery Program of the Swiss National Science Foundation under Grant 194677.

ABSTRACT Printed mathematical expression recognition (MER) models are usually trained and tested using LaTeX-generated mathematical expressions (MEs) as input and the LaTeX source code as ground truth. As the same ME can be generated by various different LaTeX source codes, this leads to unwanted variations in the ground truth data that bias test performance results and hinder efficient learning. In addition, the use of only one font to generate the MEs heavily limits the generalization of the reported results to realistic scenarios. We propose a data-centric approach to overcome this problem, and present convincing experimental results: Our main contribution is an enhanced LaTeX normalization to map any LaTeX ME to a canonical form. Based on this process, we developed an improved version of the benchmark dataset *im2latex-100k*, featuring 30 fonts instead of one. Second, we introduce the real-world dataset *realFormula*, with MEs extracted from papers. Third, we developed a MER model, *MathNet*, based on a convolutional vision transformer, with superior results on all four test sets (*im2latex-100k*, *im2latexv2*, *realFormula*, and *InftyMDB-1*), outperforming the previous state of the art by up to 88.3%.

INDEX TERMS Data-centric AI, deep learning, labeling, document analysis, mathematical expression recognition, pattern recognition.

I. INTRODUCTION

Recognizing mathematical expressions (MEs) in images and converting them into a machine-understandable format is known as mathematical expression recognition (MER). Creating a dependable MER would unlock possibilities for producing innovative tools, such as the ability to digitize, search, extract, and enhance the accessibility of mathematical equations in documents [1].

However, despite recent progress in the field of MER, it remains a challenge for two main reasons. Firstly, MEs contain many symbols, i.e., multiple alphabets, numerals, operators, and parentheses. Secondly, structural information (for example, nested superscripts and subscripts) is crucial for correctly recognizing MEs [2], [3].

In addition, we have identified a third challenging aspect that needs to be addressed. The machine-understandable format used in many MER models can cause unwanted variation. For instance, LaTeX is a popular format used by many MER models [4]. However, LaTeX allows authors to write the same ME with different LaTeX code as shown in Figure 1. Accordingly, many LaTeX commands are redundant or can be neglected without altering the canonical form or even without changing the visual appearance of an ME. For example, we observed that of the 500 different tokens in the printed MER benchmark dataset *im2latex-100k* [5], 174 tokens or 34.8% of the vocabulary is redundant or does not influence the canonical form of the ME. This leads to detrimental variability in the training data and

The associate editor coordinating the review of this manuscript and approving it for publication was Siddhartha Bhattacharyya.

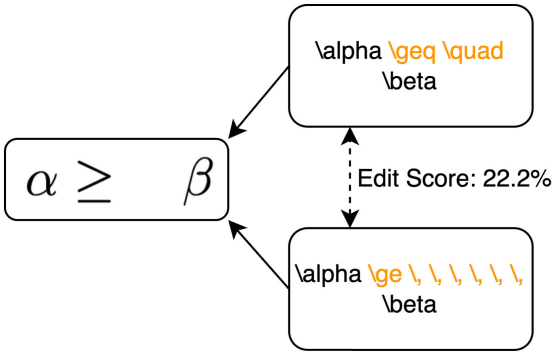


FIGURE 1. An example of an ME image that can be produced with more than one LaTeX code. While the two presented LaTeX codes are quite different (22.2% Edit score), they create the same image.

therefore to inefficient learning, excessive training data needs, and, finally, suboptimal recognition performance due to unresolved ambiguity in the model [6], [7]. Finally, the use of a single font for MEs in *im2latex-100k* heavily limits the generalization of performance results reported on this data set to realistic scenarios.

Reducing this variability is not only to reduce unwanted biases in test scores but is expected to have a high impact on learning quality of respective models and, hence, their performance. Recently, methods have emerged under the term data-centric AI that systematically engineer the data in order to improve overall system quality [8], [9]. The methodology is characterized by making the data a first-class citizen in the development process of any machine-learning-based system, thus shifting the focus away from merely manipulating the model architecture [10]. In this paper, we adopt a data-centric approach by proposing a systematic process to map an ME to a canonical LaTeX representation. Since we develop our methods to make MEs in PDFs accessible, we focus in this work on printed MER, but our approaches are also applicable to handwritten MEs and we expect similar benefits for complex handwritten MEs. We present the following major contributions: 1) A LaTeX normalization process that maps LaTeX MEs to a canonical form. 2) *Im2latexv2*, an upgraded version of *im2latex-100k* with multiple fonts and a canonical ground truth (GT). 3) *realFormula*, a real-world test set for MER. 4) Our MER model *MathNet* which outperforms the previous state of the art on all four test sets by up to 88.3%.

The remainder of this paper is structured as follows: we present related work in Section II. We discuss the issues with using LaTeX for MER in Section III. We introduce the datasets we have developed, the metrics used, and our printed-MER model in Section IV. We will then present the results of our experiments in Section V and discuss them in Section VI. Finally, we offer concluding remarks in Section VII.

II. RELATED WORK

MER has been a research task for over 50 years [11], and it still remains open. Although the focus of the MER research field has shifted to the recognition of online and offline handwritten MEs in the last years, research on printed MEs is still important to make it applicable in practice. The two fields of MER research overlap, but there are also two major differences. First, the offline handwritten MER has the extra challenge of touching symbols, which makes it harder to separate them [3], [4]. Second, the characteristics of the benchmark datasets are different. Handwritten (offline) MER uses the CROHME datasets [12] as the benchmark, with a vocabulary of 142 tokens and, on average, 18 tokens per ME. On the other hand, the printed MER benchmark dataset *im2latex-100k* [5] has a much larger vocabulary of 500 tokens, which is 3.5 times greater than the CROHME dataset. Additionally, on average, each ME in the *im2latex-100k* dataset has 2.8 times as much tokens.

However, both MER systems comprise three stages: symbol segmentation, symbol recognition, and 2D structure analysis [4]. Classic approaches, as the Infty system [13], [14] solve these stages separately, whereas end-to-end approaches address them all at once. With recent progress in deep learning, end-to-end approaches with an encoder-decoder structure have become prevalent [15]. These systems directly map input images to a semantic text representation, e.g., LaTeX. In general, the encoder is based on convolutional layers to calculate features of the image. The decoder generally uses LSTMs [16], GRUs [17], or Transformers [18], which translate the feature inputs step-by-step into a token sequence [4].

WYGIWYS, introduced by Deng et al. [15], is one of the first end-to-end MER systems. It calculates its features using a convolutional network stacked with an RNN row encoder. The token sequence is predicted by an RNN decoder with visual attention stacked with a classifier layer. Because of the end-to-end approach, large datasets are required for training. Therefore, the authors introduced *im2latex-100k* [5], which is still the classic benchmark dataset in printed MER.

Cho et al. [19] found that the performance of the encoder-decoder network for text generation declines as the length of the sentence increases. This is particularly relevant for ME sequences, which are usually longer than sentences used in image captioning. As a result, many MER models focus on enhancing the long-distance dependence of the decoder. Various approaches have been developed to overcome this issue.

Bian et al. [20] developed a bi-directional mutual learning network based on attention aggregation. The network uses two encoders, one that processes the input left-to-right and another that processes it right-to-left. They demonstrated that this structure helps alleviate the issue of long-range dependencies in RNNs. Li et *al*. [21] introduced a method for counting symbols in handwritten MER. Their weakly supervised multi-scale counting module can be combined with most encoder-decoder frameworks, and it improves the model's robustness when the ME is complex and/or long. However, it does not solve the problem with variations in writing styles. Yan et al. [22] developed ConvMath, a printed

MER system based entirely on convolutions. They introduced a convolutional decoder to better detect the 2D relation of MEs. Markazemy et al. [23] introduced a novel reinforcement learning module to process the decoder output and refine it.

Apart from focusing on the decoder, various elements of MER have been researched. Wang et al. [24] aimed to enhance the encoder by incorporating DenseNet into printed MER. Li et *al*. [25] introduced scale augmentation and drop attention to handwritten MER to improve the model performance for various ME scales. Peng et al. [26] introduced Graph Neural Network in printed MER. However, representing an ME as a graph became popular in handwritten MER [27], but not in printed MER [4]. Singh [28] investigated the visual attention in printed MER and developed two new datasets based on *im2latex-100k*. Furthermore, there have been advancements in the development of end-to-end systems for scientific documents that can recognize not only MEs but also text and tables. However, the current leading end-to-end system from Blecher et al. [29] has an Edit distance of only 87.2%, which is lower than the best current MER systems.

However, to the best of our knowledge, the influence of undesired variations in the GT has not yet been investigated in handwritten nor printed MER.

III. DETRIMENTAL LATEX VARIATIONS

MEs have a two-dimensional structure which is different from the one-dimensional structure of natural language text. Therefore, a markup language, e.g. LaTeX, is needed to convert MEs into a natural language description. LaTeX is widely used in the scientific community for writing documents. Hence, many MEs in LaTeX exist, making it appealing for printed MER. The widely used benchmark dataset *im2latex-100k* also uses LaTeX to create the MEs for the training and test datasets. However, we discovered two problematic issues with this dataset:

1) Our analysis of the dataset revealed that the whole *im2latex-100k* dataset was created with a single font. This, on the one hand, drastically limitates the generalisation capability of the performance results reported on this dataset to realistic scenarios where MEs are printed in various different font styles, normally different than the one used for training the systems. This effect was revealed in preliminary experiments when we observed a significant decrease in the performance of all tested systems by changing the font of the test set. This effect is also apparent when we compare the performance results of the baseline models of *im2latex-100k* and *im2latexv2* (refer to Tables 4 and 5).
   To address this limitation, all MEs of the *im2latex-100k* dataset were generated in many different fonts.
2) We further discovered another detrimental effect in the GT of *im2latex-100k*: As the GT of the MEs in *im2latex-100k* was taken from real papers written by different authors, there was a large variation of the GT for semantically identical MEs, as illustrated in Figure 1. These variations have nothing to do with improved generalization capabilities to be learned or shown. To the contrary: First, it reduces the validity of performance result comparisons of the different systems if this occurs in the test dataset. Second, it is detrimental for the learning of MER systems, if it occurs in the training dataset (by teaching the model that the same input has ambiguous output, leading to reduced learning [30]). In order to minimize these meaningless variations in the GT of *im2latex-100k*, we adopted a data-centric approach to develop a new LaTeX normalization procedure. The data-centric approach involves three steps. First, the model is trained using the existing training data. Second, the performance of the trained model is evaluated to identify any error patterns. Third, these error patterns are utilized to improve the training dataset (in our case by adjusting the LaTeX normalization).

These steps are repeated until no more error patterns can be detected. During this iterative process, we have identified six problematic aspects in the GT of the *im2latex-100k* dataset: mathematical fonts, white spaces, curly brackets, sub- and superscript order, tokens, and arrays. These problematic aspects together with our proposed solutions are described in Sections III-A - III-F. We designed our normalization process to address these issues and reduce undesired variations. The normalization algorithm is publicly accessible via GitHub [31].

A. MATHEMATICAL FONTS

Using different mathematical fonts, such as bold or double bold, to indicate vectors or spaces can be challenging for MER. Recognizing these mathematical fonts is simple if only one font is used for all MEs, but it becomes challenging with multiple fonts, as shown in Figure 2. Additionally, it can be challenging to create a dataset with mathematical fonts, as not all mathematical font commands work with every font, i.e., only 16 out of 59 fonts respond to the three basic mathematical font commands (`\mathcal`, `\mathbb`, and `\boldsymbol`) for all symbols. As a result, the collected ME can contain a mathematical font command that does not influence the compiled image of the ME. To avoid this, we decided to remove all mathematical font commands, which is a simplification of the task but reduces the number of labeling errors in the GT.

B. WHITE SPACES

In LaTeX, authors can adjust the white space between two symbols using various commands (e.g., `\quad`). However, these commands are primarily defined relative to the font size, making it essential for the model to accurately detect the font size, which is influenced by the font. Additionally, multiple combinations of white space commands exist for each relative white space length. This makes it impossible for the model to predict the white space commands when multiple fonts

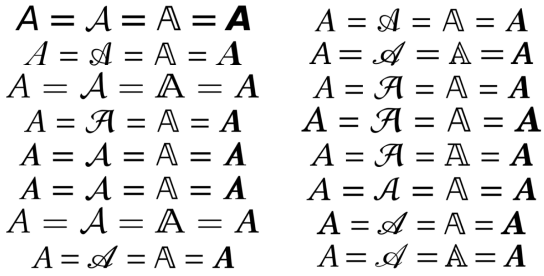


FIGURE 2. The ME: A = \mathcal{A} = \mathbb{A} = \boldsymbol{A} generated with the 16 fonts, which can render all three basic mathematical fonts.

are utilized, and the white space commands do not follow a clear pattern. Since the white space does not impact the canonical form of an ME, we decided to remove all white space commands from the GT.

C. CURLY BRACKETS

In LaTeX, curly brackets are used to define the scope of LaTeX commands. As a result, 33% of all tokens in *im2latex-100k* are curly brackets. However, the issue with curly brackets is that they are often optional and can be added without changing the visual appearance of a mathematical expression (e.g., `a_3`, `a_{3}`, and `{a_{3}}` are visually identical). Therefore, we introduced a precise definition of which curly brackets are required and which are not. This reduces ambiguity and the number of curly brackets in the GT.

D. SUB- AND SUPERSCRIPT ORDER

Symbols can have sub- and superscripts but the order in LaTeX code is irrelevant for the visual appearance of the ME. If multiple sub- and superscripts (e.g. `a^{b}_{c}^{d}`) exist, we decided to combine these to one subscript and one superscript (e.g. `a_{c}^{bd}`) to reduce ambiguity and the number of tokens. Although this normalization steps may result in errors in certain circumstances, it typically minimizes undesired variations of the GT.

E. TOKENS

We identified three issues on the token level. First, many expressions in LaTeX exist that produce the same visual symbol (e.g., `\ge` to `\geq`). Hence, we identified all redundant LaTeX expressions in *im2latex-100k* and replaced them by the canonical form. Second, some tokens in the ME imply that the ME not only contains mathematical elements (e.g., `\cite`, `\label`) or is more a graphic element than an ME (e.g., `\fbox`). Hence, we decided to delete all MEs with such tokens. Third, the tokenizer introduced by Deng et al. [15] sometimes combines two LaTeX commands in one token, e.g., the token `\right{` actually contains two tokens `\right` and `{`. This can increase the vocabulary and introduce undesired variation. To avoid this, we split up these tokens, so each token represents only one LaTeX command.

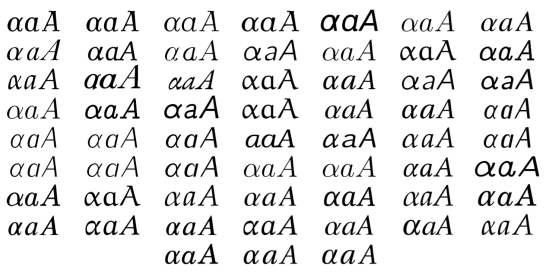


FIGURE 3. Overview of all 59 fonts in the im2latexv2 dataset.

F. ARRAYS

The array structure has the purpose to arrange elements in a grid, e.g., a matrix. However, many authors use this feature to align MEs instead (e.g., `\begin{array}{cc}a=b, & c=d\end{array}`. Additionally, the column alignment indicators (l, c, r) do not affect the semantics of the array. Moreover, not all arrays are well-defined and may contain empty columns, rows, or cells. Hence, we removed array structures used only for formatting, and reduced GT variation in the array structures by replacing all column indicators with c. We also removed sparse arrays (empty entries or number of columns doesn't match number of column alignment indicators).

IV. APPLYING LATEX NORMALIZATION FOR PRINTED MER

To evaluate our LaTeX normalization, we applied it for printed MER. Therefore, we developed an enhanced version of *im2latex-100k* described in Section IV-A, a real-world test set described in Section IV-B, and a new printed MER model described in Section IV-C. Lastly, Section IV-D gives an introduction to printed MER metrics.

A. im2latexv2

This dataset is an evolution of *im2latex-100k* and contains three major modifications over existing printed MER datasets. First, we used the normalization procedure described in Section III with minor modifications for rendering. To create controlled visual diversity, we left the column alignment indicators of arrays unchanged and did not remove the `\right` and `\left` tokens for rendering the MEs. Using the normalized MEs we can ensure that the GT and image coincide. In comparison, Deng et al. [15] used the original ME descriptions for *im2latex-100k*. Hence, the GT for the same image may vary.

Second, in contrast to *im2latex-100k*, *im2latex-90k*, and *im2latex-140k*, we rendered each ME with 30 different fonts for the training dataset and 59 for the validation and test set. The incorporation of multiple fonts makes the dataset more realistic. Furthermore, 29 fonts only appear in the validation and test set to assess a model's generalization capability. The font variation introduced this way is illustrated in Figure 3.

Third, we used 600 DPI (font size 12pt) to render the images, because down-sampling works well compared to up-sampling. In contrast, Deng et al. [15] suggested 100 DPI for the MER task. Singh [28] used 200 DPI and

TABLE 1. Overview of the reasons, why we deleted different MEs.

Category	Alg. 1	train	validation	test
im2latex-100k		75′275	8′370	10′355
white image	2	19	0	30
empty ME (corrected)	5	1 (1)	0	42 (37)
normalization step	10	882	116	179
rendering errors	15	129	11	23
im2latexv2		74′245	8′243	10′118

in handwritten MER different resolutions exist. However, the scanned images of handwritten MER correspond to resolutions of 300 to 600 DPI in printed MER. We will demonstrate the influence of the resolution on the model performance in Section V-A.

The resulting *im2latexv2* dataset contains fewer MEs than the original *im2latex-100k* due to our rendering pipeline, which includes four check criteria (see Algorithm 1 and Table 1). 19 MEs in the training set and 30 MEs in the test set had to be dropped because the image was blank. Additionally, we found 1 empty ME in the train set and 42 empty MEs in the test set. We manually corrected the empty ME in the train set and 37 MEs in the test set. We removed the other 5 empty MEs from the test set because the image depicted a drawing rather than a valid ME. Besides, our normalization step dropped 882 MEs in the training set, 116 in the validation set, and 179 MEs in the test set. The rendering step removed 129 MEs in the training set, 11 MEs in the validation set, and 23 MEs in the test set, which could not be rendered for all fonts. As a result, the training set was reduced by 1′023 MEs, the validation set by 127 MEs, and the test set by 237 MEs compared to the original *im2latex-100k*. The new normalized dataset *im2latexv2* finally contains approximately 92′600 MEs (ref. Table 1). It is publicly available on GitHub [31].

Algorithm 1 Im2latexV2 Rendering Pipeline

Require: $[F_1, I_1] \dots [F_N, I_N]$ in im2latex-100k

for $k \leftarrow 1$ to N **do**
  if I_k is *null* **then**
    continue
  end if
  if F_k is *null* **then**
    check manually
    continue
  end if
  $F_k \leftarrow$LaTeXNormalization(F_k)
  if F_k is *null* **then**
    continue
  end if
  for j, r in *renderingSetups* **do**
    $I_{k,j}, e_{k,j} \leftarrow$renderImages($F_k$, *renderingSetup*)
    if $e_{k,j}$ is not *null* **then**
      break
    end if
  end for
end for

TABLE 2. Overview of 200 randomly selected MEs. It shows various issues that arose, requiring some MEs to be excluded from the *realFormula* set.

Category	Number of images
too large	69
cropping error	9
sparse matrix	1
removed	79
correct	121

B. realFormula

By using the Mathematical Formula Detection model from Schmitt-Koopmann et al. [32], we collected over 250k ME from randomly selected arXiv papers with 600 DPI and selected 200 MEs at random for manual annotation. As shown in Table 2 we deleted 69 MEs where the image was larger than 768x2400 pixels. Nine other MEs were deleted because the image did not show the complete ME and 1 ME showed a sparse matrix. Hence, we manually annotated 121 MEs. Of these 121 MEs, 110 were single-line MEs and 11 were multi-line MEs. Five single-line MEs contained an array, and 43 MEs contained style types (`\boldsymbol`, `\mathbb`, `\mathcal`). *realFormula* is publicly available on GitHub [31].

C. MathNet

For our experiments we decided to use an encoder-decoder approach similar to the state of the art MER models.

In order to accurately process ME images, it is crucial for the encoder to extract informative features. This requires the encoder to be able to focus on small details while also considering the overall structure of the ME, such as a fraction. To handle both short-term and long-term relationships, Deng et al. [15] developed the Coarse-to-Fine Attention mechanism. However, recent advancements in image recognition have shown that vision transformers (ViTs) [33] are well-suited for this task. A further development of ViTs are convolutional vision transformers (CvTs) [34]. CvTs combine convolutions with transformers, resulting in superior performance and efficiency with a smaller model. Hence, we decided to use a CvT instead of a usual CNN encoder.

The decoder is responsible for converting the features of the encoder into the chosen markup language, i.e. LaTeX. Unlike most other MER systems, *MathNet* uses a regular decoder transformer instead of LSTMs. Vaswani et al. [18] showed that transformers are better suited for handling long sequences, as we have in printed MER. Furthermore, *im2latexv2* is much larger than *im2latex-100k*, which should benefit the training of transformers. Our decoder transformer has 8 heads and a depth of 4. On top of this, we added a classifier layer with a log softmax. An overview of our *MathNet* model with the layer sizes is shown in Fig. 4.

We used a cross-entropy loss between the GT sequence and the predicted sequence. To optimize our model, we used the Adam optimizer [35] with an initial learning rate of 0.000075 and a batch size of 36. Our model was trained on a single Nvidia Tesla V100-SXM2-32GB GPU.

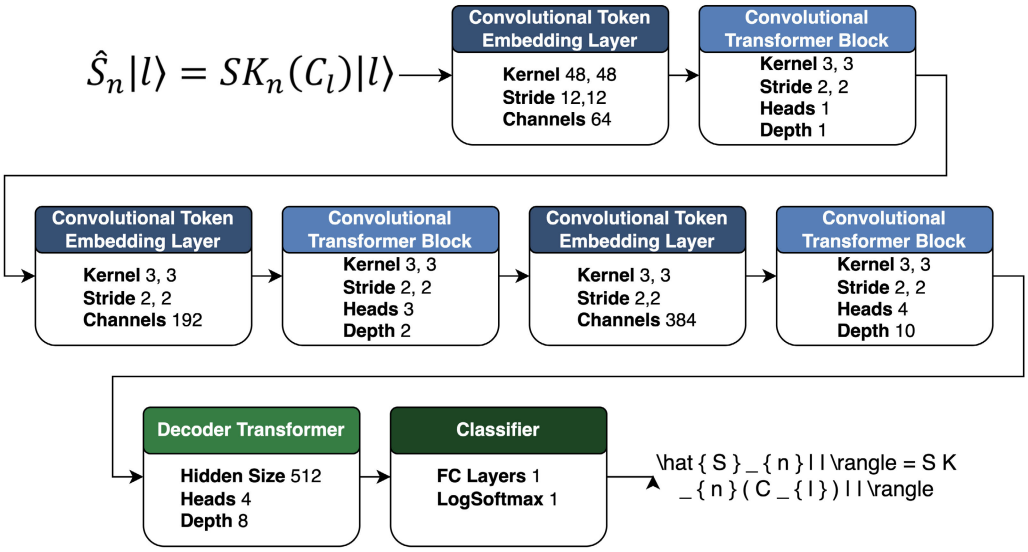


FIGURE 4. Overview of our MER model, called *MathNet*. The CvT consists of 3 layers, which are a combination of an embedding layer and a transformer block. The encoded image is decoded with a decoder transformer and a classifier layer.

In order to prevent the model from learning useless patterns in images, we applied different augmentation techniques. We have identified four patterns that the model should explicitly not learn. Firstly, to avoid confusion by white spaces (such as `\quad` and `\,`), we randomly introduced white pixel columns to the image. Secondly, we used blurring masks and changed the image resolution randomly. Thirdly, we resized the image to prevent the model from focusing on a specific text size. Lastly, we added a white border of variable size to the image to facilitate batch-wise processing and ensure that all images have the same size.

D. METRICS

Printed MER primarily uses three metrics (Edit distance, Bleu-Score, and Exact Match) to evaluate model performance. An edit distance counts the operations needed to transform one sequence into another sequence. Depending on the operations allowed, different edit distances exist. The most popular edit distance for printed MER and the one we use is the Levenshtein edit distance (*lev*). It contains three operations: 1) insert a new token, 2) delete a token, and 3) replace a token. The Edit score, as used by Deng et al., is the edit distance normalized by the max sequence length of the GT and predicted sequence (PRE) as shown in Eq. 1. For MER, an Edit score of 100% is a perfect prediction.

$$EditScore = (1 - \frac{lev(GT, PRE)}{max(len(GT), len(PRE))}) \cdot 100\% \quad (1)$$

The Bleu score compares subsequences of two sequences. A predefined number, usually 4 in MER, determines the maximum subsequence length. To determine the Bleu score you create n-grams for the sequences and then calculate the precision between the n-grams of sequences. The Bleu score is the average precision with a brevity penalty to discourage overly short predictions. However, the Bleu score is designed for longer sequences, and errors at the sequence borders count less than errors in the middle. This behavior can significantly impact the score of MEs. Exact Match measures the amount of fully correct MEs. It makes no distinction between partially correct ME and completely incorrect ME.

With respect to PDF accessibility, we regard the Edit score as the most relevant metric because it indicates the amount of work that must be manually done to correct all errors in a recognized ME. The Bleu-4 score shows unwanted behavior for short MEs and focuses more on patterns than on the correct order of the tokens. The interpretation of exact match is very limited through the binary output.

However, as discussed in Section III, semantically identical MEs can be produced with different LaTeX code sequences. Hence, the metric results are largely influenced by the used tokenizer and normalization. For example, if the x in Eq. 2 should be a 5, the Edit score would rise from 96.3% (1 of 27) to 97.8% (1 of 45) if curly brackets had been added in the GT around each entry in the array, which would mean a 40.5% reduction of the Edit error rate (1 - Edit score). The Bleu-4 score shows similar behavior.

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & x & 6 \\ 7 & 8 & 9 \end{pmatrix} \quad (2)$$

V. EXPERIMENTS WITH PRINTED MER

This section presents the results of four experiments. The first experiments (see Section V-A) demonstrate the influence of the image resolution on the model performance. Experiments two to five are comparison experiments with printed MER models. To ensure a fair comparison, we used the provided pre-trained models (*WYGIWYS*, *i2l-strips*, *i2l-nopool*) and normalized the predictions with our normalization process.

TABLE 3. Influence of the training image resolution on the *im2latexv2* test set. We trained our model with 100, 200, 300, and 600 DPI.

DPI	Edit [%]	Bleu-4 [%]	EM [%]
100	78.2	66.0	6.0
200	93.5	93.2	56.9
300	96.9	96.9	75.0
600	**98.0**	**98.2**	**84.9**

We used the following four datasets to compare the models: 1) The benchmark dataset *im2latex-100k*. 2) Our enhanced version *im2latev2* which includes multiple fonts. 3) Our developed real-world dataset *realFormula* with MEs images extracted from papers, to demonstrate how well the systems perform in a real-world environment. 4) *InftyMDB-1* contains MEs images, which were scanned with 600 DPI. This dataset is also intended to evaluate real-world performance, specifically the impact of scanning noise.

It is important to note that *i2l-strips* and *i2l-nopool* used a modified dataset built upon *im2latex-100k* which had a different split between training, validation, and testing. Hence, MEs of the test sets of *im2latex-100k* and *im2latexv2* could be in the training set of *i2l-strips* and *i2l-nopool*.

A. OPTIMAL IMAGE RESOLUTION

When the resolution is low, the image has fewer details and the model has to focus on the general structure. On the other hand, high-resolution images provide more detail which can enable the model to differentiate better between symbols.

However, there is no clear definition what the optimal image resolution for MER is (with the standard font size of 12pt). According to Deng et al. [15], 100 DPI images are recommended, while Singh [28] used images with 200 DPI. In contrast, handwritten MER mainly uses image sizes that correspond to resolutions between 300 and 600 DPI [12].

We trained our model on various image resolutions to demonstrate the impact of this resolution, as shown in Table 3. We used 100, 200, 300, and 600 DPI image resolutions. The images of the test set were scaled accordingly. Our results reveal a significant improvement between 100 and 200 DPI. Moreover, the model's performance still improves with even higher resolutions. However, we did not test resolutions higher than 600 DPI because 600 DPI is typically the maximum for scanned documents. For the subsequent experiments, we used the model with 600 DPI training images.

B. im2latex-100k

This Section presents the results on the *im2latex-100k* test set. *im2latex-100k* contains images with 100 DPI. However, *i2l-strips* and *i2l-pool* were trained with 200 DPI images, and our model was trained with 600 DPI images so they require larger images. We used two techniques to create larger images. First, we resized the original images with OpenCV to the training size. Second, we rendered the original MEs without normalization using a LaTeX environment to eliminate the influence of insufficient resolution.

TABLE 4. Results of the *im2latex-100k* test set. We run the models once with the original images, resized to the training size, and once with the images rendered with the optimal resolution.

Model	Edit [%]	Bleu-4 [%]	EM [%]
WYGIWYS	88.6	90.3	**78.6**
I2l-strips (resized)	32.5	12.7	0
I2l-strips (rendered)	86.9	86.1	76.3
I2l-nopool (resized)	32.0	13.4	0
I2l-nopool (rendered)	86.8	85.9	76.2
MathNet (our) (resized)	88.6	86.0	31.8
MathNet (our) (rendered)	**94.7**	**94.5**	63.4

As shown in Table 4, rendering an image with a higher resolution achieves better results as resizing the original images. *MathNet* achieved the same Edit scores (88.6%) as *WYGIWYS* with the resized images. However, the Edit error rate (1 - Edit score) nearly halved from 11.4% to 5.3% when the images were rendered with 600DPI. *i2l-strips* and *i2l-nopool* performed poorly with the resized images (32.5% and 32.0%), but similarly to *WYGIWYS* with the rendered images (86.9% and 86.8%). Interestingly, the exact match score of *MathNet* is low compared to the other systems. Hence, the fewer errors of *MathNet* must be more widely spread over the different MEs than those of the other systems.

C. im2latexv2

This section presents the results with the *im2latexv2* test set. We assigned a random font for each ME in the test set. We used the same font-ME combination for all models to avoid influencing the results by using different fonts for the same ME. Since *im2latexv2* uses 600 DPI images, we resized the images for *i2l-strips*, *i2l-nopool*, and *WYGIWYS* to the training image resolution. As presented in Table 5, *WYGIWYS*'s Edit score drops dramatically from 88.6% to 37.2% compared to *im2latex-100k*. However, *i2l-strips* and *i2l-nopool* handle multiple fonts better, with only a small decrease of 11 pp. and 10.8 pp. in the Edit score. In contrast, our model shows a 2.5 pp. increase in the Edit score. We attribute this increase to the fact that *im2latex-100k* includes problematic mathematical fonts, as explained in Section VI-A4.

D. REALFORMULA

This section presents the results of the *realFormula* test set. Table 6 provides an overview of the results. The table shows that our model reaches an Edit score of 88.3%. This is about three times higher than *WYGIWYS* (27.5%) and approximately one-third higher than *i2l-strips* (65.1%) and *i2l-nopool* (65.2%). In order to quantify the impact of multi-line formulae we have split the MEs into multi-line (M) and single-line (S) MEs as discussed in Section VI-A5. To determine the influence of the array element we have filtered out all MEs with the token elements `\begin{array}` and `\end{array}` (nA), which is discussed in Section VI-A3. Additionally, we have filtered out all MEs with mathematical fonts (nMF); this issue is discussed in Section VI-A4.

TABLE 5. Results of the im2latexv2 test set. We resized the images to the optimal size. Errors is the summed Levenshtein distance over all MEs. Array errors is the summed Levenshtein distance of all MEs with an array structure. nA is the edit score of all MEs without an array structure.

Model	Train Dataset	Edit [%]	Bleu-4 [%]	EM [%]	Errors	Array Errors	Edit nA [%]
WYGIWYS	im2latex-100k	37.2	23.9	0	564′700	31′742	37.1
I2l-strips	im2latex-140k	75.9	65.9	10.3	143′802	28′539	79.2
I2l-nopool	im2latex-140k	76.0	66.4	10.4	144′860	28′015	78.9
MathNet (our)	im2latexv2	**97.2**	**96.8**	**83.9**	**16′596**	**8′728**	**98.6**
MathNet (our)	im2latex-100k	78.2	65.9	10.3			
MathNet (our)	im2latexv2 (vanilla font)	90.4	84.3	26.4			

TABLE 6. Edit scores [%] of the *realFormula* test set. S: single line ME, M: multi-line ME, nA: no arrays, A: arrays, nMF: no mathematical fonts, MF: mathematical fonts.

Model	all	S	S nA	S A	S nMF	S MF	M
WYGIWYS	27.5	28.6	29.6	11.6	28.9	27.8	22.5
I2l-strips	65.1	76.6	81.7	17.7	82.6	65.1	15.9
I2l-nopool	65.2	77.1	81.9	20.7	83.5	65.0	13.9
MathNet (our)	**88.3**	**92.5**	**93.3**	**84.1**	**94.1**	**89.5**	**71.2**

TABLE 7. Prediction results on the InftyMDB-1 dataset (scanned MEs).

Model	Edit [%]	Bleu-4 [%]	EM [%]
WYGIWYS	17.3	7.1	0
I2l-strips	63.5	46.5	4.1
I2l-nopool	63.2	46.4	3.4
MathNet (our)	**89.2**	**85.4**	**35.4**

TABLE 8. Prediction results for MEs in the *realFormula* test set with mathematical fonts.

Mathematical Font Type	Edit [%]	Bleu-4 [%]	EM [%]
`\boldsymbol`	98.4	96.4	63.6
`\mathcal`	90.4	83.7	18.2
`\mathbb`	84.6	75.4	0.0
`\operatorname`	83.4	75.7	0.0

E. InftyMDB-1

This Section presents the results on the *InftyMDB-1* test set [36]. *InftyMDB-1* contains 4400 images of scanned MEs with a resolution of 600 DPI. We used the pandoc library to covert the MathML GT into LaTeX GT and processed the resulting LaTeX strings similar to the other datasets.

As shown in Table 7, the resulting performance of *MathNet* is about the same compared to *realFormula* test set. However, it demonstrates that *MathNet* is not significantly affected by the noise of the scanning process. In contrast, the performance of *WYGIWYS*, *i2l-strips*, and *i2l-nopool* drops by 10.2 pp., 2.0 pp., and 1.6 pp.. This highlights that these models are probably affected by the noise of the scanning process. However, since our focus is on scientific PDFs, we assume that scientific PDFs are usually available in native digital format. Hence, scanned documents with geometric deformation, coloration, and noise are considered as not in our research focus.

VI. DISCUSSION

A. DATA RELATED ACHIEVEMENTS AND CHALLENGES

As our results reveal, our data-centric approach with the LaTeX normalization and augmentation process is very beneficial for the training of robust printed MER models. The influence of our normalization and the use of multiple fonts on the model performance is discussed in Section VI-A1. Section VI-A2 demonstrates that our model is adept at working with fonts not included in the training set.

However, in our error analysis, we encountered two significant challenges. First, the array element was the main culprit of errors, as detailed in Section VI-A3. Second, the absence of mathematical fonts and multi-line MEs in the *im2latexv2* training dataset poses a challenge for our model on the *realFormula* test set, as discussed in Section VI-A4 and Section VI-A5. Section VI-A6 gives an overview of the most frequent token errors with MathNet and *im2latexv2*.

1) THE IMPACT OF NORMALIZATION AND MULTIPLE FONTS

We conducted experiments to separately analyse the influences of our model architecture, our normalization process, and the use of multiple fonts. We trained the model three times, once with the *im2latex-100k* dataset, once with the *im2latexv2* dataset using only the basic font, and once with the full *im2latexv2* dataset. The results are shown in Table 5.

When we used the *im2latex-100k* dataset, our model's Edit score (78.2%) was more than double that of *WYGIWYS* (37.2%) and was 2.3 pp. 2.2 pp. higher than *i2l-strips* and *i2l-nopool*. This demonstrates the beneficial network design of our model. The advantage of our model architecture is analyzed further in Section VI-B. However, the normalization process has a much stronger impact on the model's Edit score, with a 12.2 pp. improvement when using the *im2latexv2* dataset with the vanilla font for all MEs. The remaining 6.8 pp. improvement is explained by the use of multiple fonts for the MEs during training. In summary, the model architecture is marginally better as state of the art model architectures. However, two-thirds of the improvement compared to state of the art models are due to our LaTeX normalization process, while the remaining third is attributed to the use of multiple fonts. This reveals the significant influence of our LaTeX normalization process during model training and, hence, the value of the new dataset *im2latexv2*.

2) NON-TRAINING FONTS

The *im2latexv2* training set only includes 30 of the 59 fonts in the test set. We tested our model's ability to work with fonts not in the training set and found that that the font influence is

TABLE 9. Analysis of the Levenshtein operations required to correct the *MathNet* predictions on the *im2latexv2* test set. The table shows the 10 most frequent tokens required to be inserted or deleted and the 10 most frequent pairs of tokens that must be replaced by the other.

Insert		Delete		Replace		
Token	Count	Token	Count	Token 1	Token 2	Count
}	2078	}	666	*	\ast	196
{	2000	{	589	^	\star	95
^	678	_	122	a	\alpha	79
_	678	*	96	\nu	v	74
2	345	^	63	\phi	\varphi	71
)	344	-	49	\rangle	>	42
-	271	1	47	\dots	\cdots	40
(	263	2	46	\epsilon	\varepsilon	36
1	242	&	43	\langle	<	34
,	196	c	42	\psi	\Psi	29

negligible. Specifically, we achieved a 97.5% Edit score for MEs with fonts in the training set and 96.8% for MEs with fonts that were not. Overall, this demonstrates our model's strong generalizability.

3) ARRAY ISSUE

LaTeX users normally use the array structure to create a matrix (`\begin{array}` ⋯ `\end{array}`), but some authors use it to format their ME instead. To address this unwanted variation we introduced normalization steps in Section III to reduce the use of the array environment for formatting purposes. However, even with our normalization process, the array structure remains challenging for MER, as shown in Table 5. Out of all the prediction errors on the *im2latexv2* test set with our model, 52.6% are related to MEs with an array structure. However, this array structure is only present in 4.8% of all the MEs. Therefore, by removing MEs that use the array structure, our model's Edit error rate is reduced by 50% (from 2.8% to 1.4%). *i2l-strips* and *i2l-nopool* also see reductions from 24.1% to 20.8% and from 24% to 21.1%, respectively. The effect on *WYGIWYS* is not significant. We attribute this to the high overall Edit error rate of *WYGIWYS*. The same problems with array structures can be seen in the results of Table 6 for the *realFormula* test set. For instance, *MathNet* achieves an Edit score of 93.3% for single line MEs without arrays and 84.1% for single line MEs with arrays.

4) MATHEMATICAL FONT ISSUE

In ME, changes to the font style of symbols (mathematical fonts) are used to indicate, e.g., vectors and spaces. As these mathematical fonts are not rendered correctly for all fonts, we decided to remove all mathematical font tokens in *im2latev2* to ensure the images are rendered correctly. Consequently, *MathNet* cannot detect mathematical fonts. Our results with the *realFormula* test set reveal that removing MEs with mathematical fonts in the training set has a significant influence on the model's real-world performance. Without counting the mathematical font tokens as an error, the Edit score of MEs without mathematical fonts is 94.1% (column S nMF), whereas MEs rendered with mathematical fonts, it drops to only 89.5% (column S MF). Table 8 shows that MEs rendered with the three mathematical fonts `\mathcal`, `\mathbb`, and `\operatorname` are especially challenging for *MathNet*. In contrast, the mathematical font `\boldsymbol` has no negative influence on the performance. Nevertheless, mathematical fonts are a limitation of *MathNet* and *im2latexv2* and, hence, the predicting results of *MathNet* deteriorate for MEs containing mathematical fonts. This issue is to be addressed in future research.

5) MULTI-LINE ME

The MEs in the *im2latex-100k* dataset are limited to 150 tokens, so there are almost no multi-line MEs. However, in the *realFormula* dataset we had to drop 69 MEs because they were too large, and these were all multi-line MEs. Together with the 11 multi-line MEs in the final dataset, 80 of the original 200 MEs were multi-line MEs (see Table 2). This reveals that a real-world MER needs to handle multi-line MEs in addition to single-line MEs.

Table 6 shows that our model performs 30% better in terms of Edit score for single-line MEs (92.5%) compared to multi-line MEs (71.2%). This may be because our training set, *im2latexv2*, consists primarily of single-line expressions. The other models suffer a much more dramatic performance drop for multi-line MEs to 14-23% Edit score. Furthermore, by using a straightforward y-cut algorithm, we can strongly improve our model's performance for multi-line MEs from 71.2% to 96.2% Edit score. As a result, when the y-cut algorithm performs well, we can robustly recognize multi-line MEs even with our model mainly trained on single-line MEs.

6) MOST FREQUENT TOKEN ERRORS

To better understand the open challenges of our *MathNet* model, we analyzed the Levenshtein operations needed to correct the predictions. Table 9 shows the 10 most frequent tokens that needed be corrected. It is not surprising that the curly brackets are the primary culprit of errors because they are the most frequent tokens in the GT. Also, the sub- and superscript tokens (_ and ^) are still tricky for our model, even after our normalization step.

The replace operations reveal that the model is mainly confused by visually very similar symbols. However, their occurrences are small compared to the number of errors with curly brackets.

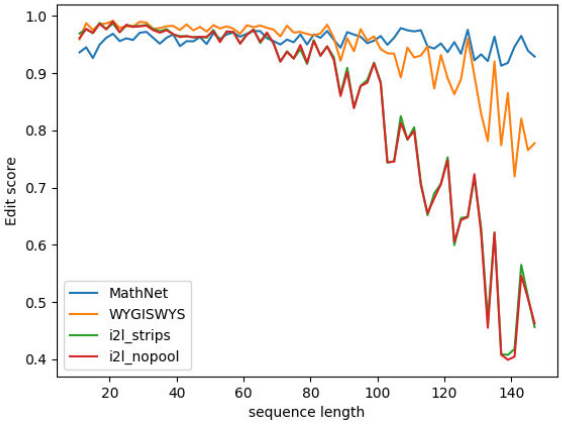


FIGURE 5. The plot shows the average edit score per sequence length for the different models and the *im2latex-100k* dataset. The x-axis shows the number of tokens in the ME with a bin width of 3. The y-axis shows the average Edit score of each bin. A perfect prediction has a edit score of 1.

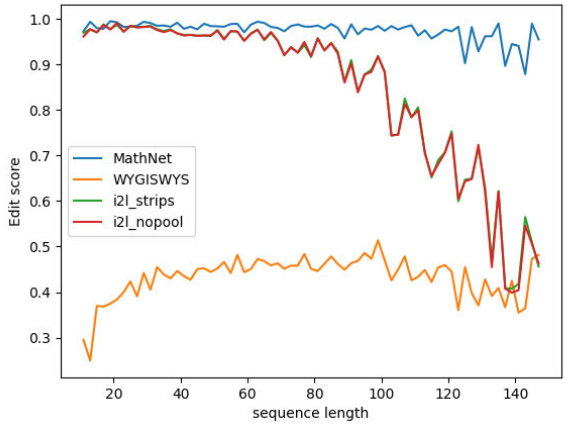


FIGURE 6. The plot shows the average edit score per sequence length for the different models and the *im2latexv2* dataset. The x-axis shows the number of tokens in the ME with a bin width of 3. The y-axis shows the average edit score of each bin. A perfect prediction has a edit score of 1.

B. MODEL RELATED ACHIEVEMENTS AND CHALLENGES

As discussed in Section II, many MER models employ LSTMs with specialized mechanisms to improve long-distance learning. We addressed this issue using a transformer architecture. Our analysis, depicted in Figures 5 and 6, shows that the Edit score of *MathNet* does not decrease with the sequence length of the MEs, indicating that transformers are effective in learning long-distance relationships in MEs.

VII. CONCLUSION

We introduced the novel printed MER model *MathNet*, incorporating a CvT encoder and transformer decoder. *MathNet* achieves outstanding results for *im2latex-100k* (Edit score: 94.7%), *im2latexv2* (Edit score: 97.2%), *realFormula* (Edit score: 88.3%), and *InftyMDB-1* (Edit score: 89.2%), reducing the Edit error rate to the prior state of the art for these datasets by 53.5% (from 11.4% to 5.3%), 88.3% (from 24% to 2.8%), 66.4% (from 34.8% to 11.7%), and 70.4% (from 36.5% to 10.8%), respectively. These results were achieved with our transformer-based model architecture and on an inherently data-centric approach normalizing and augmenting the training data. We found that detrimental variations in the LaTeX GT of *im2latex-100k* exist. To reduce this undesired variations, we proposed a LaTeX normalization method. Our LaTeX normalization process enables the model to focus on the canonical form of an ME instead of learning non-relevant variations. We demonstrated that our LaTeX normalization process is mainly responsible for the model's superior performance. Moreover, we introduced an augmented dataset, *im2latexv2*, an enhanced and normalized version of *im2latex-100k* with multiple fonts, and *realFormula* which contains annotated real ME images from arXiv papers. We also showed that a simple y-cut algorithm can expand single-line MER to multi-line MER.

Despite promising effectiveness, the Edit scores of all models investigated were significantly lower on *realFormula* and *InftyMDB-1* compared to *im2latex-100k* and *im2latexv2*, which indicates a difference between synthetic (*im2latex-100k* and *im2latexv2*) and real-world datasets (*realFormula* and *InftyMDB-1*). The removal of mathematical fonts styles in *im2latexv2*, such as bold and italics, limits the correct recognition of MEs that use these mathematical fonts styles in *realFormula*. An extended version of *im2latexv2* with mathematical fonts could solve this problem. Additionally, the correct cutting of ME lines heavily supports multi-line ME recognition, making stable line detectors a precondition.

After testing the handwritten benchmark dataset *CROHME* with our model *MathNet* and our LaTeX normalization, we could not find evidence that our LaTeX normalization process helps to improve the recognition performance. We think this is because the characteristics of *CROHME* and *im2latex-100k* are vastly different. The MEs in *CROHME* are on average only one-third as long as in *im2latex-100k*, and the vocabulary is significantly smaller, consisting of only 142 tokens compared to 500 tokens in printed MER. As a result, our LaTeX normalization only reduces the original 142 tokens to 121 (canonical) tokens, which is much less than with *im2latex-100k*. Furthermore, the MEs in *CROHME* are simpler and do not contain arrays, mathematical fonts, and other complex elements. This leads to the conclusion that the detrimental variation in *CROHME* is much lower than in *im2latex-100k*. However, we believe that for more complex handwritten MEs, our LaTeX normalization process could be as beneficial as it is for printed MER.

Generative pretrained transformers with multimodal input have shown significant progress in image recognition. However, testing a few ME images with GPT-4 from OpenAI indicates that the results, although impressive, have not yet reached the state of the art in MER. Nevertheless, combining generative AI with MER could be a promising approach worth exploring.

For our upcoming research steps, we plan to combine *FormulaNet* [32] and *MathNet* to develop a semi-automatic

captioning system for MEs in PDFs. With this system, we expect to significantly improve the accessibility of PDFs specifically for MEs and also enable easy searching and extracting of MEs from PDFs.

REFERENCES

[1] F. M. Schmitt-Koopmann, E. M. Huang, and A. Darvishy, "Accessible PDFs: Applying artificial intelligence for automated remediation of STEM PDFs," in *Proc. 24th Int. ACM SIGACCESS Conf. Comput. Accessibility*. New York, NY, USA: Association for Computing Machinery, Oct. 2022, pp. 1–6, doi: 10.1145/3517428.3550407.

[2] A. Belaid and J.-P. Haton, "A syntactic approach for handwritten mathematical formula recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-6, no. 1, pp. 105–111, Jan. 1984, doi: 10.1109/TPAMI.1984.4767483.

[3] K.-F. Chan and D.-Y. Yeung, "Mathematical expression recognition: A survey," *Int. J. Document Anal. Recognit.*, vol. 3, no. 1, pp. 3–15, Aug. 2000.

[4] R. Aggarwal, S. Pandey, A. K. Tiwari, and G. Harit, "Survey of mathematical expression recognition for printed and handwritten documents," *IETE Tech. Rev.*, vol. 39, no. 6, pp. 1245–1253, Nov. 2022, doi: 10.1080/02564602.2021.2008277.

[5] A. Kanervisto, Jun. 2016, "im2latex-100k," *Zenodo*, doi: 10.5281/zenodo.56198.

[6] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning (still) requires rethinking generalization," *Commun. ACM*, vol. 64, no. 3, pp. 107–115, Feb. 2021, doi: 10.1145/3446776.

[7] D. Arpit, S. Jastrzebski, N. Ballas, D. Krueger, E. Bengio, M. S. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio, and S. Lacoste-Julien, "A closer look at memorization in deep networks," in *Proc. Int. Conf. Mach. Learn.*, Jul. 2017, pp. 233–242.

[8] DeepLearningAI. (Mar. 2021). *A Chat With Andrew on MLOps: From Model-Centric to Data-Centric AI*. [Online]. Available: https://www.youtube.com/watch?v=06-AZXmwHjo

[9] P.-P. Luley, J. M. Deriu, P. Yan, G. A. Schatte, and T. Stadelmann, "From concept to implementation: The data-centric development process for AI in industry," in *Proc. 10th IEEE Swiss Conf. Data Sci. (SDS)*, Jun. 2023, pp. 73–76, doi: 10.1109/sds57534.2023.00017.

[10] T. Stadelmann, T. Klamt, and P. H. Merkt, "Data centrism and the core of data science as a scientific discipline," *Arch. Data Sci. Ser. A*, vol. 8, no. 2, p. 16, 2022, doi: 10.5445/IR/1000143637.

[11] R. H. Anderson, "Syntax-directed recognition of hand-printed two-dimensional mathematics," in *Proc. Assoc. Comput. Machinery Inc. Symp. Interact. Syst. Experim. Appl. Math.* NY, USA: Association for Computing Machinery, Aug. 1967, pp. 436–459, doi: 10.1145/2402536.2402585.

[12] Y. Xie, H. Mouchère, F. S. Liwicki, S. Rakesh, R. Saini, M. Nakagawa, C. T. Nguyen, and T.-N. Truong, "ICDAR 2023 CROHME: Competition on recognition of handwritten mathematical expressions," in *Proc. Document Anal. Recognit. (ICDAR)*, in Lecture Notes in Computer Science, G. A. Fink, R. Jain, K. Kise, and R. Zanibbi, Eds. Cham, Switzerland: Springer, 2023, pp. 553–565, doi: 10.1007/978-3-031-41679-8.

[13] M. Suzuki, F. Tamari, R. Fukuda, S. Uchida, and T. Kanahori, "INFTY: An integrated OCR system for mathematical documents," in *Proc. ACM Symp. Document Eng.* New York, NY, USA: Association for Computing Machinery, Nov. 2003, pp. 95–104, doi: 10.1145/958220.958239.

[14] C. Malon, S. Uchida, and M. Suzuki, "Mathematical symbol recognition with support vector machines," *Pattern Recognit. Lett.*, vol. 29, no. 9, pp. 1326–1332, Jul. 2008, doi: 10.1016/j.patrec.2008.02.005.

[15] Y. Deng, A. Kanervisto, J. Ling, and A. M. Rush, "Image-to-markup generation with coarse-to-fine attention," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, Sydney, NSW, Australia, Aug. 2017, pp. 980–989.

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[17] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Doha, Qatar, Oct. 2014, pp. 1724–1734, doi: 10.3115/v1/d14-1179.

[18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30. Glasgow, U.K.: Curran Associates, 2017, pp. 1–11.

[19] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder–Decoder approaches," in *Proc. 8th Workshop Syntax, Semantics Struct. Stat. Transl.*, Doha, Qatar. Stroudsburg, PA, USA: Association for Computational Linguistics, Oct. 2014, pp. 103–111, doi: 10.3115/v1/w14-4012.

[20] X. Bian, B. Qin, X. Xin, J. Li, X. Su, and Y. Wang, "Handwritten mathematical expression recognition via attention aggregation based bi-directional mutual learning," in *Proc. AAAI Conf. Artif. Intell.*, Jun. 2022, vol. 36, no. 1, pp. 113–121, doi: 10.1609/aaai.v36i1.19885.

[21] B. Li, Y. Yuan, D. Liang, X. Liu, Z. Ji, J. Bai, W. Liu, and X. Bai, "When counting meets HMER: Counting-aware network for handwritten mathematical expression recognition," in *Proc. 17th Eur. Conf. Comput. Vis. (ECCV)*, Tel Aviv, Israel. Berlin, Germany: Springer, Oct. 2022, pp. 197–214, doi: 10.1007/978-3-031-19815-1.

[22] Z. Yan, X. Zhang, L. Gao, K. Yuan, and Z. Tang, "ConvMath: A convolutional sequence network for mathematical expression recognition," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Milan, Italy, Jan. 2021, pp. 4566–4572, doi: 10.1109/ICPR48806.2021.9412913.

[23] A. Mirkazemy, P. Adibi, S. M. S. Ehsani, A. Darvishy, and H.-P. Hutter, "Mathematical expression recognition using a new deep neural model," *Neural Netw.*, vol. 167, pp. 865–874, Oct. 2023, doi: 10.1016/j.neunet.2023.08.045.

[24] J. Wang, Y. Sun, and S. Wang, "Image to latex with DenseNet encoder and joint attention," *Proc. Comput. Sci.*, vol. 147, pp. 374–380, Jan. 2019, doi: 10.1016/j.procs.2019.01.246.

[25] Z. Li, L. Jin, S. Lai, and Y. Zhu, "Improving attention-based handwritten mathematical expression recognition with scale augmentation and drop attention," in *Proc. 17th Int. Conf. Frontiers Handwriting Recognit. (ICFHR)*, Sep. 2020, pp. 175–180, doi: 10.1109/ICFHR2020.2020.00041.

[26] S. Peng, L. Gao, K. Yuan, and Z. Tang, "Image to LaTeX with graph neural network for mathematical formula recognition," in *Proc. 16th Int. Conf. Document Anal. Recognit.*, Lausanne, Switzerland. Berlin, Germany: Springer, Sep. 2021, pp. 648–663, doi: 10.1007/978-3-030-86331-9_42.

[27] M. Mahdavi, R. Zanibbi, H. Mouchere, C. Viard-Gaudin, and U. Garain, "ICDAR 2019 CROHME + TFD: Competition on recognition of handwritten mathematical expressions and typeset formula detection," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sep. 2019, pp. 1533–1538, doi: 10.1109/ICDAR.2019.00247.

[28] S. S. Singh, "Teaching machines to code: Neural markup generation with visual attention," 2018, *arXiv:1802.05415*.

[29] L. Blecher, G. Cucurull, T. Scialom, and R. Stojnic, "Nougat: Neural optical understanding for academic documents," 2023, *arXiv:2308.13418*.

[30] N. Simmler, P. Sager, P. Andermatt, R. Chavarriaga, F.-P. Schilling, M. Rosenthal, and T. Stadelmann, "A survey of un-, weakly-, and semi-supervised learning methods for noisy, missing and partial labels in industrial vision applications," in *Proc. 8th Swiss Conf. Data Sci. (SDS)*, Jun. 2021, pp. 26–31.

[31] F. Schmitt-Koopmann. (2023). *MathNet: A Data-Centric Approach for Printed Mathematical Expression Recognition*. [Online]. Available: https://github.com/felix-schmitt/MathNet

[32] F. M. Schmitt-Koopmann, E. M. Huang, H.-P. Hutter, T. Stadelmann, and A. Darvishy, "FormulaNet: A benchmark dataset for mathematical formula detection," *IEEE Access*, vol. 10, pp. 91588–91596, 2022, doi: 10.1109/ACCESS.2022.3202639.

[33] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Represent.*, May 2021, pp. 1–21.

[34] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, and L. Zhang, "CvT: Introducing convolutions to vision transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 22–31, doi: 10.1109/ICCV48922.2021.00009.

[35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

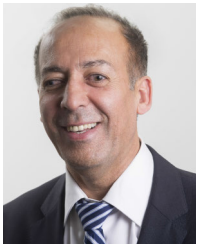
[36] A. Fujiyoshi, M. Suzuki, and S. Uchida. (2009). *InftyMDB-1*. [Online]. Available: https://www.inftyproject.org/download/inftydb/InftyMDB-1.zip

FELIX M. SCHMITT-KOOPMANN received the B.Sc. degree in mechanical engineering and the M.Sc. degree in robotics, systems, and control from ETH Zürich, in 2019 and 2021, respectively. He is currently pursuing the Ph.D. degree with the People and Computing Laboratory, University of Zürich. He is also a member of the Institute of Applied Informatics, Zurich University of Applied Sciences (ZHAW). His research interests include accessibility, AI, and document analysis.

ELAINE M. HUANG received the Ph.D. degree from the College of Computing, Georgia Institute of Technology, in 2006. She is currently a Professor of human–computer interaction with the Department of Informatics, University of Zürich (UZH), where she leads the People and Computing Research Group. Prior to joining UZH, in 2010, she was a Researcher at Motorola Labs and a Professor with the Department of Computer Science, University of Calgary. Her research interests include the use of technology to address issues of inequality and other societal challenges.

HANS-PETER HUTTER (Member, IEEE) received the Ph.D. degree in technical science from ETH Zürich (ETHZ), in 1997, with a focus on hybrid HMM/ANN approaches to speech recognition over telephone lines. He studied electrical engineering with ETHZ. He joined UBS Ubilab as a Postdoctoral Researcher, where he worked on a European project for HMM-based speaker identification over the telephone. At the same time, he was a Co-Lecturer at ETHZ in two speech processing modules. In 1997, he joined Zurich University of Applied Sciences (ZHAW), Winterthur, where he was a Professor in computer science on various projects in the area of speech recognition and user-centered design of graphical and voice user interfaces. In 2005, he founded the Institute of Applied Information Technology (InIT), ZHAW School of Engineering, together with his colleagues and was the head of the institute, until 2010. At the same time, he was also the Head of the Human-Information Interaction Group, InIT, which he is still leading today.

THILO STADELMANN (Senior Member, IEEE) received the D.Sc. degree from Marburg University, Germany, in 2010, with a focus on multimedia analysis and voice recognition. He studied computer science in Giessen and Marburg. He is currently a Professor of AI/ML with the ZHAW School of Engineering, Winterthur, Switzerland, the Director of the ZHAW Centre for Artificial Intelligence, and the Head of the Machine Perception and Cognition Group. He held engineering and leadership roles in the automotive industry for several years prior to his appointment at ZHAW. His current research interests include robust deep learning to solve diverse pattern recognition tasks, such as document analysis or industrial or medical computer vision.



ALIREZA DARVISHY is currently a Professor of ICT accessibility and the Head of the ICT Accessibility Laboratory, Zurich University of Applied Sciences (ZHAW), Switzerland. He serves as an Independent Reviewer for European research projects, such as the Active Assisted Living (AAL) Program. He is a Principal Investigator of the "Accessible Scientific PDFs for All" project, funded by the Swiss National Science Foundation.