Textual Description for Mathematical Equations

Ajoy Mondal and C V Jawahar
Centre for Visual Information Technology,
International Institute of Information Technology, Hyderabad, India,
Email: ajoy.mondal@iiit.ac.in and jawahar@iiit.ac.in

***Abstract*—Reading of mathematical expression or equation in the document images is very challenging due to the large variability of mathematical symbols and expressions. In this paper, we pose reading of mathematical equation as a task of generation of the textual description which interprets the internal meaning of this equation. Inspired by the natural image captioning problem in computer vision, we present a mathematical equation description (MED) model, a novel end-to-end trainable deep neural network based approach that learns to generate a textual description for reading mathematical equation images. Our MED model consists of a convolution neural network as an encoder that extracts features of input mathematical equation images and a recurrent neural network with attention mechanism which generates description related to the input mathematical equation images. Due to the unavailability of mathematical equation image data sets with their textual descriptions, we generate two data sets for experimental purpose. To validate the effectiveness of our MED model, we conduct a real-world experiment to see whether the students are able to write equations by only reading or listening their textual descriptions or not. Experiments conclude that the students are able to write most of the equations correctly by reading their textual descriptions only.**

***Keywords*-Mathematical symbols; mathematical expressions; mathematical equation description; document image; convolution neural network; attention; recurrent neural network.**

I. Introduction

Learning of mathematics is necessary for students at every stage of student life. Solving mathematical problems is an alternative way to develop and improve their mathematical skills. Unfortunately, blind and visually impaired (VI) students face the difficulties to especially learn mathematics due to their limitations in reading and writing mathematical formulas. In generally, human readers help those categories of students to access and interpret materials or documents of mathematics courses. However, at every time, it is impossible and impractical for those categories of students having human reader; because of the cost and the limited availability of the trained personnel. Braille is the popular and more convenient way to access the document for blind and VI students. Unfortunately, many documents are not available in Braille, since the conversion of mathematical documents in Braille is more difficult and complicated [1]. Moreover, it is also difficult for students who are comfortable for reading literary Braille transcriptions [2].

Other than Braille, sound based representation of documents is also an important and popular way to access information for those kinds of students. In this direction, DAISY books and talking books are commonly used audio materials for understanding documents. However, these books are less prepared for mathematical expressions or equations (MEs) [2]. Recently, text-to-speech (TTS) systems have been widely used by blind and VI students to read electronic text through computers. TTS systems convert digital text into synthetic speech. Unfortunately, most available TTS systems can read only plain text. They fail to generate appropriate speech when it comes across mathematical equations.

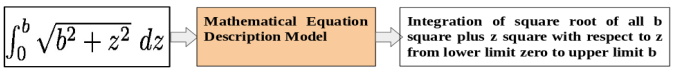


Figure 1: Our model treats reading of mathematical equation in a document image as generation of textual description which interprets the internal meaning of this equation.

Many researchers realized that it is very important to enhance the accessibility of mathematical materials to the blind and VI students and developed TTS based mathematical expressions reading systems [3]–[5]. Some of these developed systems need extra words to read an ME. Due to the extra words, the rendered audio is very long. Hence, the students may not always be able to interpret the main point of the expressions due to long audio duration. Moreover, most of these existing automatic math reading systems take MEs as input in the form of LaTeX or other similar markup languages. Unfortunately, all available mathematical documents are not in the form of LaTeX or any other markup language. Therefore, generation of LaTeX or other markup languages corresponds to mathematical documents is also challenging [2].

In this paper, our goal is to develop a framework, called as mathematical equation description (MED) model which can help the blind and VI students for reading/interpreting internal meaning of MEs present in the document images. We pose reading of ME as a problem of generation of natural language description. Basically, our proposed MED model automatically generates textual (natural language) description which can able to interpret the internal meaning of the ME. For examples, $\int sin\ x\ dx$ is a ME and its textual description is like "integration of $sin\ x$ with respect to x".

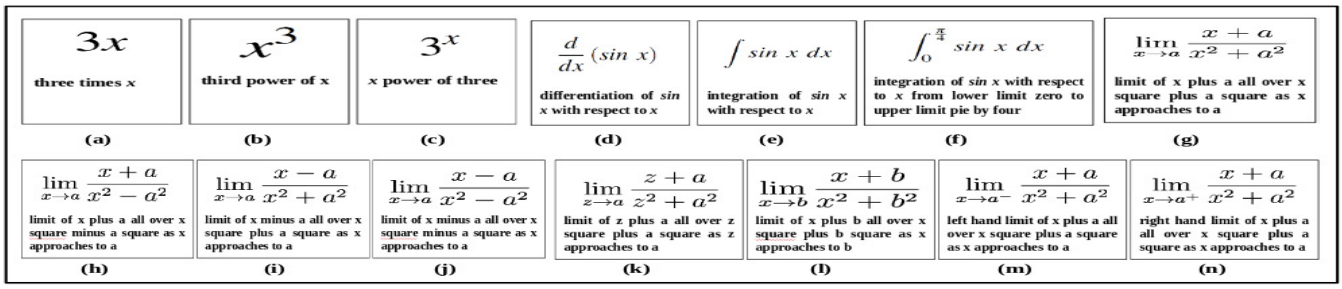


Figure 2: Example of sensitivity of variables, operators and their positions while reading the equations. Only '3' in (a) is changing position in (b), 'x' in (a) is changing position in (c), 'differentiation operator' in (d) is changed by 'integration' in (e), and 'differentiation operator' in (d) is changed by 'finite integration' in (f), '$+$' operator in denominator of (g) is changed by '$-$' in (h), '$+$' operator in nominator of (g) is changed by '$-$' in (i), '$+$' operators in both nominator and denominator of (g) are changed by '$-$' in (j), variable 'x' in (g) is changed by 'z' in (k), constant 'a' in (g) is changed by 'b' in (l), limit value 'a' in (g) is replaced by limit value 'a^-' in (m), and limit value 'a' in (g) is changed to 'a^+'in (n).

With the textual description, the blind and VI students can easily read/interpret the MEs. Figure 1 shows the generated textual description of an MEI using our MED model. This task is closely related to image description/captioning task [6]. However, description of equation is very sensitive with respect to variables, operators and their positions. Figure 2 illustrates the sensitivity of variables, operators and their positions during generation of textual description for MEs. For example, $3x$ in Figure 2(a), it can be read as "three times x". While 3 is changing its position e.g. 3^x in Figure 2(c), textual sentence "x power of three" for reading, it is totally different from previous equation. As per our understanding goes, this is the first work where reading/interpreting of MEs is posed as a generation of textual description task.

The main inspiration of our work comes from image captioning, a recent advancement in computer vision. In this paper, we propose an end-to-end trainable deep network to generate natural language descriptions for the MEs which can read/interpret the internal meaning of these expressions. The network consists of two modules: encoder and decoder. The encoder encodes the ME images using Convolution Neural Networks (CNNs). Long Short Term Memory (LSTM) network as decoder that takes the intermediate representation to generate textual descriptions of corresponding ME images. The attention mechanism impels the decoder to focus on specific parts of the input image. The encoder, decoder and attention mechanism are trained in a joint manner. We refer this network as Mathematical Equation Description (MED) network.

In particular, our contributions are as follows.

- We present an end-to-end trainable deep network with the combination of both vision and language models to generate description of MEs for reading/interpreting MEs .
- We generate two data sets with ME images and their corresponding natural language descriptions for our experimental purpose.
- We conduct a real world experiment to establish effectiveness of the MED model for reading/interpreting mathematical equation.

II. Related Work

A. Mathematical Expression Recognition

Automatic recognition of MEs is one of the major tasks towards transcribing documents into digital form in the scientific and engineering fields. This task mainly consists of two major steps: symbol recognition and structural analysis [7]. In case of symbols recognition, the initial task is to segment the symbols and then to recognize the segmented symbols. Finally, structural analysis of the recognized symbols have been done to recognize the mathematical expressions. These two problems can be solved either sequentially [8] or a single framework (global) [9]. However, both of these sequential and global approaches have several limitations including (i) segmentation of mathematical symbols is challenging for both printed and handwritten documents as it contains a mix of text, expressions and figures; (ii) symbols recognition is difficult because a large number of symbols, fonts, typefaces and font sizes [7]; (iii) for structural analysis, commonly two dimensional context free grammar is used which requires prior knowledge to define math grammar [10] and (iv) the complexity of parsing algorithm increases with the size of math grammar [11].

Due to the success of deep neural network in computer vision task, the researchers adopted deep neural models to recognize mathematical symbols [12], [13] and expressions [14]–[18]. In [12], [13], the authors considered CNN along with bidirectional LSTM to recognize mathematical symbols. Whereas, [14], [15] explored the use of attention based image-to-text models for generating structured markup for MEs. These models consist of a multi-layer convolution

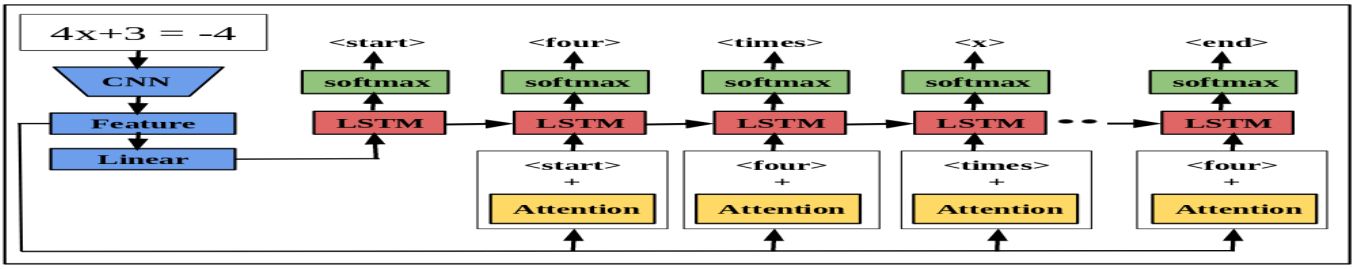


Figure 3: Overview of mathematical expression description network. Our model uses a end-to-end trainable network consisting of CNN followed by a language generating LSTM. It generates textual description of an input mathematical expression image in natural language which interprets its internal meaning.

network to extract image features with an attention based recurrent neural network as a decoder for generating structured markup text. In the same direction, Zhang *et al.* [18] proposed a novel end-to-end approach based on neural network that learns to recognize handwritten mathematical expressions (HMEs) in a two-dimensional layout and produces output as one-dimensional character sequences in the LaTeX format. Here, the CNN, as encoder is considered to extract feature from HMEs images and a recurrent neural network is employed as decoder to generate LaTeX sequences.

B. Image Captioning

Image captioning is a task which automatically describes the content of an image using properly formed English sentences. Although, it is a very challenging task, it helps the visually impaired people to better understand the content of images on the web. Recently, a large variety of deep models [6], [19]–[22] have been proposed to generate textual description of natural images. All these models considered recurrent neural network (RNN) as language models conditioned on the image features extracted by convolution neural networks and sample from them to generate text. Instead of generating caption for whole image, a handful of approaches to generate captions for image regions [6], [23], [24]. In contrast of generating a sentence, various models have also been been introduced to generate paragraph for describing content of the images in literature [25], [26] by considering a hierarchy of language models.

III. Mathematical Equation Description

A. Overview

Our MED model takes a mathematical expression image (MEI) as an input and generates a natural language sentence to describe the internal meaning of this expression. Figure 3 provides an overview of our model. It consists of encoder and decoder networks. The encoder extracts deep features to richly represent the equation images. The decoder uses the intermediate representation to generate a sentence to describe the meaning of the ME. The attention mechanism impels the decoder to focus on specific parts of the input image. Each of these networks are discussed in details in the following subsections.

B. Feature Extraction using Encoder

The MED model takes a MEI and generates its textual description $\mathbf{Y}$ encoded as a sequence of 1-of-K encoded words.

$$\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_T\},\ \mathbf{y}_i \in R^K \tag{1}$$

where K is the size of the vocabulary and T is the length of the description. We consider a Convolution Neural Network (CNN) as an encoder in order to extract a set of feature vectors. We assume that the output of CNN encoder is a three-dimensional array of size $H \times W \times D$, and consider the output as a variable length grid of L vectors, $L = H \times W$ as referred to annotation vectors. Each of these vector is D-dimensional representation that corresponds to a local region of the input image.

$$\mathbf{A} = \{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_L\},\ \mathbf{a}_i \in R^D \tag{2}$$

We extract features from a lower convolution layer in order to obtain a correspondence between the feature vectors and regions of the image. This allows the decoder to selectively focus on certain regions of the input image by selecting a subset of all these feature vectors.

C. Sentence Generation using Decoder

We employ LSTM [27] as a decoder that produces a sentence by generating one word at every time step conditioned on a context vector $\hat{\mathbf{z}}_t$, the hidden state $\mathbf{h}_t$ and the previously generated word $\mathbf{y}_{t-1}$. It produces word at time step t using the following equation:

$$p(\mathbf{y}_t|\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_{t-1}, \mathbf{x}) = f(\mathbf{y}_{t-1}, \mathbf{h}_t, \hat{\mathbf{z}}_t), \tag{3}$$

where $\mathbf{x}$ denotes the input MEI and f denotes a multi-layered perceptron (MLP) which is expanded in Eq. (7). The hidden state $\mathbf{h}_t$ of LSTM is computed using following equation:

$$
\begin{aligned}
\mathbf{i}_t &= \sigma(\mathbf{W}_{yi}\mathbf{E}\mathbf{y}_{t-1} + \mathbf{U}_{hi}\mathbf{h}_{t-1} + \mathbf{V}_{zi}\hat{\mathbf{z}}_t) \\
\mathbf{f}_t &= \sigma(\mathbf{W}_{yf}\mathbf{E}\mathbf{y}_{t-1} + \mathbf{U}_{hf}\mathbf{h}_{t-1} + \mathbf{V}_{zf}\hat{\mathbf{z}}_t) \\
\mathbf{o}_t &= \sigma(\mathbf{W}_{yo}\mathbf{E}\mathbf{y}_{t-1} + \mathbf{U}_{ho}\mathbf{h}_{t-1} + \mathbf{V}_{zo}\hat{\mathbf{z}}_t) \\
\mathbf{g}_t &= \tanh((\mathbf{W}_{yc}\mathbf{E}\mathbf{y}_{t-1} + \mathbf{U}_{hc}\mathbf{h}_{t-1} + \mathbf{V}_{zc}\hat{\mathbf{z}}_t) \\
\mathbf{c}_t &= \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t \\
\mathbf{h}_t &= \mathbf{o}_t \odot \tanh(\mathbf{c}_t).
\end{aligned} \tag{4}
$$

Here, $\mathbf{i}_t$, $\mathbf{f}_t$, $\mathbf{c}_t$, $\mathbf{o}_t$ and $\mathbf{h}_t$ are the input, forget, memory, output and hidden states of the LSTM, respectively. The vector $\hat{\mathbf{z}}_t$ is a context vector which captures visual information of a particular image region. The context vector $\hat{\mathbf{z}}_t$ (in Eq. (4)) is a dynamic representation of the relevant part of the input image at time step t. We consider soft attention defined by Bahdannu *et al.* [28] which computes weight α_{ti} of each annotation vectors $\mathbf{a}_i$ conditioned on the previous LSTM hidden state $\mathbf{h}_{t-1}$. Here, we parameterize attention as MLP which is jointly trained:

$$
\begin{aligned}
e_{ti} &= \mathbf{v}_a^T \tanh(\mathbf{W}_a\mathbf{h}_t + \mathbf{U}_a\mathbf{a}_i) \\
\alpha_{ti} &= \frac{exp(e_{ti})}{\sum_{k=1}^{L} exp(e_{tk})}.
\end{aligned} \tag{5}
$$

Let $n^{'}$ be the dimension of the attention, then $\mathbf{v}_a \in R^{n^{'} \times n}$, $\mathbf{U}_a \in R^{n^{'} \times D}$. After computation of the weights α_{ti}, the context vector $\hat{\mathbf{z}}_t$ is calculated as follows:

$$
\hat{\mathbf{z}}_t = \sum_{i=1}^{L} \alpha_{ti}\mathbf{a}_i. \tag{6}
$$

This weight α_{ti} will make decoder to know which part of input image is the suitable place to attend to generate the next predicted word and then assign a higher weight to the corresponding annotation vectors $\mathbf{a}_i$. m and n denote the dimensions of embedding and LSTM, respectively; $E \in R^{m \times K}$ is the embedding matrix. σ is sigmoid activation function and $\odot$ is element wise multiplication.

Finally, the probability of each predicted word at time t is computed by the context vector $\hat{\mathbf{z}}_t$, current LSTM hidden state $\mathbf{h}_t$ and previous predicted word $\mathbf{y}_{t-1}$ using the following equation:

$$
p(\mathbf{y}_t|\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_{t-1},\ \mathbf{x}) = g(\mathbf{W}_o(\mathbf{E}\mathbf{y}_{t-1} + \mathbf{W}_h\mathbf{h}_t + \mathbf{W}_z\hat{\mathbf{z}}_t)), \tag{7}
$$

where g denotes a softmax activation function over all the words in the vocabulary; $\mathbf{E}$, $\mathbf{W}_o \in R^{K \times n}$, $\mathbf{W}_h \in R^{m \times n}$ and $\mathbf{W}_c \in R^{m \times D}$ are learned parameters initialized randomly.

The initial memory state $\mathbf{c}_0$ and hidden state $\mathbf{h}_0$ of the LSTM are predicted by an average of the annotation vectors fed through two separate MLPs ($f_{init,\ c}$, $f_{init,\ h}$)

$$
\begin{aligned}
\mathbf{c}_0 &= f_{init,\ c}\left(\frac{1}{L}\sum_{i=1}^{L}\mathbf{a}_i\right) \\
\mathbf{h}_0 &= f_{init,\ h}\left(\frac{1}{L}\sum_{i=1}^{L}\mathbf{a}_i\right).
\end{aligned} \tag{8}
$$

Expression	Description
$10x$	ten times x
x^2	x square or second power of x
$\sqrt{x}$	second root of x
$\frac{x}{10}$	x over ten
$(x+y)^2$	second power of all x plus y
$log_2 x$	log x to base two
$\frac{x}{y}$	x over y
$\frac{x}{y+z}$	x over y plus z
$\frac{x+y}{z}$	x plus y all over z
$\frac{x^z}{y^2}$	x over y square
$x + \frac{y}{z}$	x all plus y over z
$\frac{x-y}{y+z}$	x minus y all over y plus z
$\frac{x^2}{y}$	x square over y
$\frac{x-y}{z} + t$	x minus y all over z all plus t
$e^{(1+x)}$	exponential of all one plus x
$e^x + 1$	exponential of x all plus one
$e^{(1+x)} - 1$	exponential of all one plus x all minus one
$\int x\ dx$	integral of x with respect to x
$\int_0^1 x\ dx$	integral of x with respect to x from lower limit zero to upper limit one
$\lim_{x \to 0} \frac{sin\ x}{x}$	limit of $sin\ x$ over x as x approaches to zero
$\frac{d}{dx}(x^2)$	differentiation of x square with respect to x
$x - 6 = 3$	x minus six equal to three
$x + 7 > 10$	x plus seven greater than ten
$x + 2y = 7,$	x plus two times y equal to seven and
$x - y = 3$	x minus y equal to three

Table I: Natural language phrases to uniquely describe mathematical equations.

D. Implementation Details

Pre-processing: Textual descriptions of the MEs are pre-processed with basic tokenization algorithm by keeping all words that appeared at least 4 times in the training set.

Training Details: The training objective of our MED model is to maximize the predicted word probability as given in Eq. (7). We use cross entropy as the objective function:

$$
O = -\sum_{t=1}^{T} \log p(y_t^{gt}|\mathbf{y}_t,\ \mathbf{x}), \tag{9}
$$

where y_t^{gt} represents the ground truth word at time step t.

We consider 299K images and their corresponding textual descriptions to train the model. We consider pre-trained ResNet-152 model [29] (on ImageNet [30]). We do this in all the experiments. We train the network with batch size of 50 for 60 epochs. We use stochastic gradient descent (SGD) with fixed learning rate 10^{-4}, momentum $= 0.5$ and weight decay $= 0.0001$. All the weights were randomly initialized except for the CNN. We use 512 dimensions for the embedding and 1024 for the size of the LSTM memory. We consider a dropout layer after each convolution layer and set as 0.5. The best trained model is determined in terms of BLEU-4 score on validation set. For further implementation and

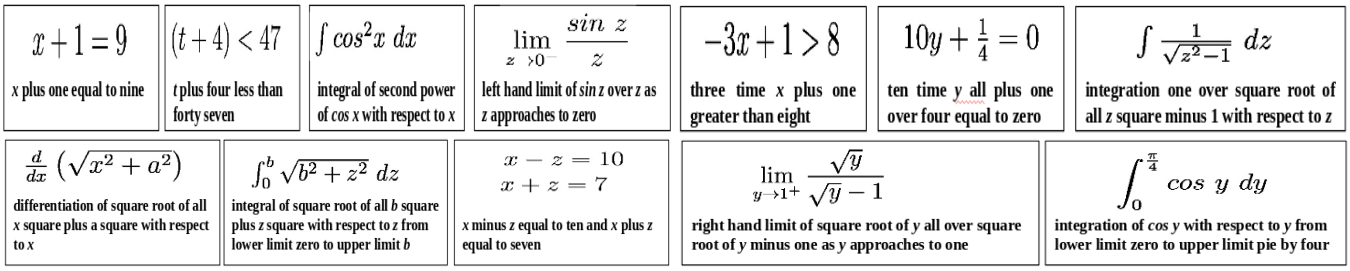


Figure 4: Few sample MEI and their corresponding textual description of `Math-Exp-Syn` data set.

architecture details, please refer to the source code at: *https://github.com/ajoymondal/Equation-Description-PyTorch*.

Decoding: In decoding stage, our main aim is to generate a most likely textual description for a given MEI:

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} \ \log p(\mathbf{y}|\mathbf{x}). \tag{10}$$

Different from training procedure, the ground truth of previous predicted word is not available. We employ beam search [31] of size 20 during decoding procedure. A set of 20 partial hypothesis beginning with the start-of-sentence <start> is maintained. At each time step, each partial hypothesis in the beam is expanded with every possible word. Only the 20 most likely beams are kept. When the <start> token is encounter, it is removed from the beam and added to the set of complete hypothesis. This process is repeated until the output word becomes a symbol corresponding to the end-of-sentence <end>.

IV. Data Sets and Evaluation Metrics

A. Data Sets

Unavailability of mathematical equation image data sets and their textual descriptions inspired us to generate data sets for experimental purpose. Various issues must be concerned during generation of unambiguous textual description of a mathematical equation. One important issue is that the same textual description can lead to the different expressions. For example, the textual description like "x plus y over z" could be description of two possible equations: either $\frac{x+y}{z}$ or $x + \frac{y}{z}$. Thus, an algorithm should be carefully designed to generate an unambiguous textual description corresponds to exactly one expression. As per our knowledge goes, no mathematical expression data sets with their textual descriptions is available for experiment. We create a data set, referred as `Math-Exp-Syn` with large number of synthetically generated MEIs and their descriptions. For this purpose, we create sets of predefined functions (e.g. linear equation, limit, etc.), variables (e.g. x, y, z, etc.), operators (e.g. $+$, $-$, etc.) and constants (e.g. 10, 1, etc.) and sets of their corresponding textual descriptions. We develop a python code which randomly selects a function, variable, operator and constant from the corresponding predefined sets and automatically generates mathematical equation as an image and corresponding textual description in the text format. We make our `Math-Exp-Syn` data generation code available at: *https://github.com/ajoymondal/Equation-Description-PyTorch*. We also create another data set, referred as `Math-Exp` by manually annotating a limited number of MEIs. During creation of both these data sets, we take care the uniqueness of the equations and their descriptions. We consider the following natural language sentences listed in table I to uniquely describe the internal meaning of the equations.

Data set	Division	No. images							Total
		LE	IE	PLE	LT	DI	IN	FIN	
`Math-Exp-Syn`	Training	41K	43K	43K	44K	39K	40K	36K	299K
	Validation	5K	5K	5K	6K	4K	5K	4K	37K
	Test	5K	5K	5K	6K	4K	5K	4K	37K
`Math-Exp`	Test	1K	0.64K	0.05K	0.68K	0.06K	0.2K	0.1K	2.7K

Table II: Category level statistics of considered data sets. LE: linear equation, IE: inequality, PLE: pair of linear equations, LT: limit, DI: differentiation, IN: integral and FIN: finite integral.

In this work, we limit ourselves to only seven categories of MEs: *linear equation*, *inequality*, *pair of linear equations*, *limit*, *differentiation*, *integral* and *finite integral*. Table II displays the category wise statistics of these data sets. Figure 4 shows few sample images and their descriptions of `Math-Exp-Syn` data set.

B. Evaluation Metrics

In this work, we evaluate the generated descriptions for MEI with respect to three metrics: BLEU [32], CIDEr [33], and ROUGE [34] which are popularly used in natural language processing (NLP) and image captioning tasks. All these metrics basically measure the similarity of a generated sentence against a set of ground truth sentences written by humans. Higher values of all these metrics indicate that the generated sentence (text) is more similar to the ground truth sentence (text).

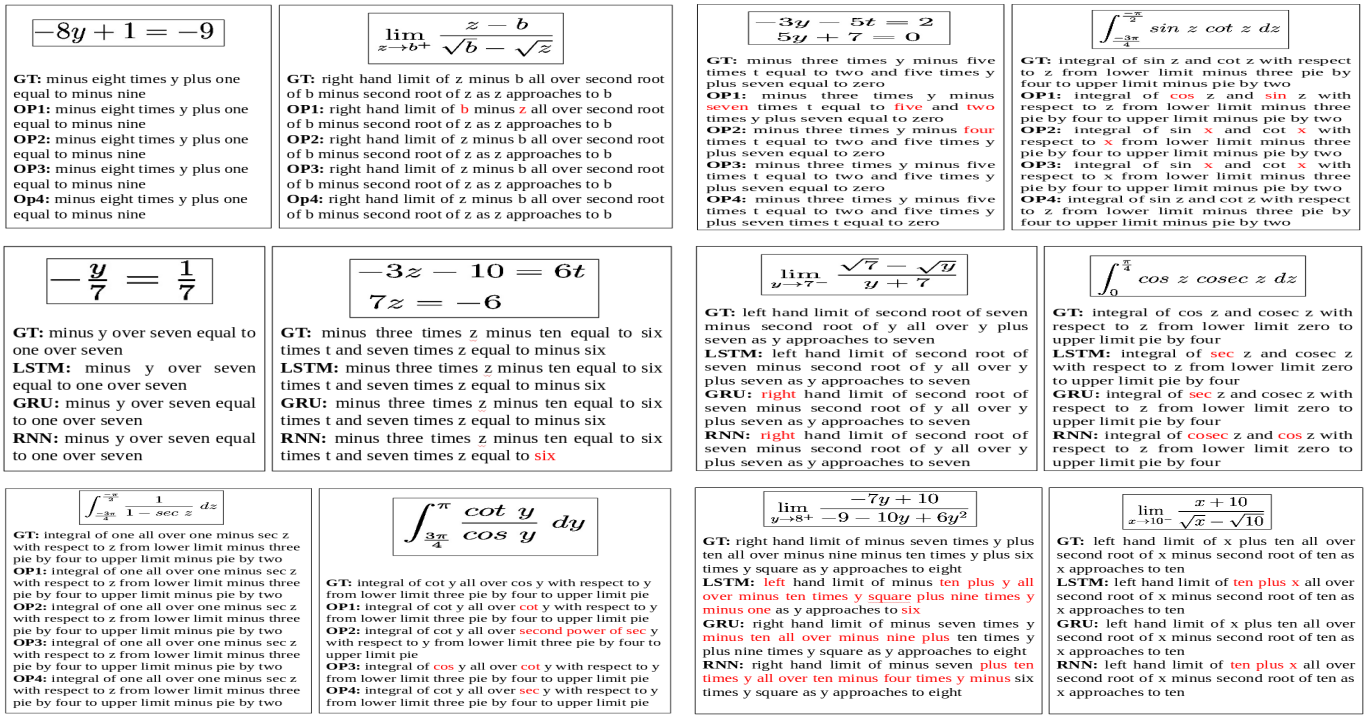


Figure 5: Visual illustration of sample results of test `Math-Exp-Syn` data set produced by our MED framework. **GT**: ground truth description, **OP1**: description generated by ResNet152+LSTM, **OP2**: description generated by ResNet152+LSTM+Attn., **OP3**: description generated by ResNet152†+LSTM and **OP4**: description generated by ResNet152†+LSTM+Attn., **LSTM**: description generated by ResNet152†+LSTM+Attn., **GRU**: description generated by ResNet152†+GRU+Attn., **RNN**: description generated by ResNet152†+RNN+Attn., † indicates fine-tune and Attn. denotes attention in decoder. Red colored text indicates wrongly generated text.

V. Experiments and Results

An extensive set of experiments is performed to assess the effectiveness of our MED model using several metrics on the ME data sets.

A. Ablation Study

A number of ablation experiments is conducted to quantify the importance of each of the components of our algorithm and to justify various design issues in the context of mathematical equation description. We use `Math-Exp-Syn` data set for this purpose.

Pre-trained Encoder: It is well known that the deeper networks are beneficial for the large scale image classification task. We conduct an experiment with different depths of the pre-trained models to analyze their performances on the mathematical equation description task. Detailed scores of equation description using the various pre-trained models are listed in Table III.

Fine-tuned vs. Without Fine-tuned Encoder and Attention vs. Without Attention in Decoder: The considered encoder, ResNet-152 pre-trained on ImageNet [30] is not effective without fine-tuning due to domain heterogeneity (natural images and MEIs). We perform an experiment to establish potency of fine-tuning on the equation description task. The attention mechanism tells the decoder to focus on a particular region of the image while generating the description related to that region of the image. We do an experiment to analyze the effectiveness of attention mechanism on the mathematical equation description task. Observation of the experiments is quantitatively reported in

Models	Test Performance					
	B-1	B-2	B-3	B-4	C	R
ResNet-18†+LSTM+Attn.	0.971	0.949	0.927	0.906	0.960	9.014
ResNet-34†+LSTM+Attn.	0.973	0.952	0.931	0.910	0.962	9.058
ResNet-50† +LSTM+Attn.	0.978	0.959	0.940	0.922	0.968	9.172
ResNet-101† +LSTM+Attn.	0.979	0.960	**0.941**	**0.923**	0.968	9.179
ResNet-152† +LSTM+Attn.	**0.981**	**0.962**	**0.941**	**0.923**	**0.971**	**9.184**

Table III: It illustrates that the deeper pre-trained model gets better representation and improves textual description accuracy with respect to three evaluation measures: **BLEU**-1 (**B**-1), **BLEU**-2 (**B**-2), **BLEU**-3 (**B**-3), **BLEU**-4 (**B**-4), **CIDEr** (**C**) and **ROUGE** (**R**). Number along with the model refers to the depth of the corresponding model. '†' denotes that the encoder is fine-tuned during training. LSTM with attention is considered as a decoder.

Models	Test Performance					
	B-1	B-2	B-3	B-4	C	R
ResNet-152+LSTM.	0.976	0.956	0.937	0.918	0.965	9.156
ResNet-152+LSTM+Attn.	0.977	0.956	0.937	0.918	0.966	9.163
ResNet-152†+LSTM.	0.978	0.960	0.940	0.922	0.968	9.182
ResNet-152† +LSTM+Attn.	**0.981**	**0.962**	**0.941**	**0.923**	**0.971**	**9.184**

Table IV: Quantitative illustration of effectiveness of fine-tuning the encoder and attention in decoder during training on MED task. '†' denotes fine-tune.

Table IV. This table highlights the effectiveness of fine-tune and attention in mathematical equation description task. First row of Figure 5 visually illustrates the effectiveness of fine-tuning the pre-trained ResNet-152 and LSTM with attention for MED task.

RNN *vs.* GRU *vs.* LSTM*:* We also conduct an experiment to analyze performances of LSTM, Gated Recurrent Units (GRU) and Recurrent Neural Networks (RNN) on generating captions for mathematical equation images. In this experiment, we consider pre-trained ResNet-152 as an encoder which is fine-tuned during training and different decoders: RNN, GRU and LSTM with attention mechanism. Table V displays the numerical comparison between three decoder models. The table highlights that LSTM is more effective than other two models for mathematical equation description task. Second and third rows of Figures 5 display the visual outputs. This figure highlights that LSTM is able to generate text most similar to the ground truth.

Models	Test Performance					
	B-1	B-2	B-3	B-4	C	R
ResNet-152†+RNN+Attn.	0.977	0.958	0.939	0.920	0.967	9.179
ResNet-152†+GRU+Attn.	0.979	0.959	0.939	0.920	0.968	9.182
ResNet-152† +LSTM+Attn.	**0.981**	**0.962**	**0.941**	**0.923**	**0.971**	**9.184**

Table V: Performance comparison between RNN, GRU and LSTM with attention mechanism on the mathematical equation description task. We fine-tune the encoder during training process.

Models	Data sets	Division	Scores					
			B-1	B-2	B-3	B-4	C	R
MED	Math-Exp-Syn	test set	0.981	0.962	0.941	0.923	0.971	9.184
MED	Math-Exp	test set	0.975	0.956	0.936	0.917	0.966	9.146

Table VI: Quantitative results of our MED model on standard evaluation metrics for both Math-Exp-Syn and Math-Exp data sets. Both the cases MED is trained using training set of Math-Exp-Syn data set.

B. Quantitative Analysis of Results

The quantitative results obtained using our MED model for both Math-Exp-Syn and Math-Exp data sets are listed in Table VI.

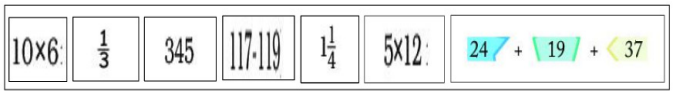


Figure 6: Sample cropped mathematical equation images from NCRT class V mathematical book for real world experiment.

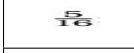
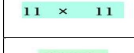
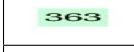
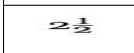
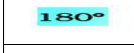
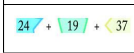
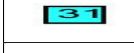
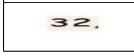
Original cropped image	Textual description generated by MED given to Students	Equation Written by Students
	five by sixteen	$\frac{5}{16}$
	eleven into eleven	11×11
	three hundred and sixty three	363
	two and one by two	$2\frac{1}{2}$
	one hundred and eighty degree	180^o
	eight plus nine plus three	$8 + 9 + 3$
	one thousand three hundred and eleven	1131
	thirty two point	32.

Table VII: Summary of real world experiments. **First Column:** cropped equation images. **Second Column:** textual descriptions generated by the MED and given to the students and ask them to write corresponding equations by reading the descriptions. **Third Column:** equations written by the students.

C. Real world Experiments

We conduct a real world experiment to see whether the students are able to write the equations by only reading or listening their textual descriptions or not. For this purpose, we create a test set of mathematical equation images which are cropped from NCRT class V mathematical book[1]. Test set consists of 398 cropped equation images of various types of equations: integer, decimal, fraction, addition, subtraction, multiplication and division. Figure 6 shows the sample cropped mathematical equation images from NCRT class V mathematical book. Our MED system generates the textual description for each of these equations. The list of descriptions of equations is given to the students and ask to write the corresponding mathematical equations within 1 hour. Twenty students participate in this test. If anyone of the students writes the incorrect equations by only reading or listening their textual descriptions. Then the answer is wrong otherwise correct. Among 398 equations, students are able to correctly write 359 equations within time by reading their textual descriptions generated by our MED model. For remaining 39 equations, our MED model generates wrong descriptions due to the presence of other structural elements

[1]https://www.ncertbooks.guru/ncert-maths-books/

(i.e. triangle, square, etc). Table VII highlights the few results of this test. Since, descriptions generated by MED model are wrong, students write wrong equations by reading wrongly generated descriptions. From this test, we conclude that our MED model is effective for reading equations by generating their textual descriptions.

VI. Conclusions

In this paper, we introduce a novel mathematical equation description (MED) model for reading mathematical equations for blind and visually impaired students by generating textual descriptions of the equations. Unavailability of mathematical images and their textual descriptions, inspires us to generate two data sets for experiments. Real-world experiment concludes that the students are able to write mathematical expression by reading or listening their descriptions generated by the MED network. This experiment establishes the effectiveness of the MED framework for reading mathematical equation for the blind and VI students.

References

[1] V. Moço and D. Archambault, "Automatic conversions of mathematical braille: A survey of main difficulties in different languages," in *ICCHP*, 2004.

[2] W. Wongkia, K. Naruedomkul, and N. Cercone, "i-math: Automatic math reader for thai blind and visually impaired students," *Computers & Mathematics with Applications*, 2012.

[3] N. Soiffer, "MathPlayer: web-based math accessibility," in *International ACM SIGACCESS Conference on Computers and Accessibility*, 2005.

[4] R. D. Stevens, A. D. Edwards, and P. A. Harling, "Access to mathematics for visually disabled students through multimodal interaction," *HCI*, 1997.

[5] S. Medjkoune, H. Mouchere, S. Petitrenaud, and C. Viard-Gaudin, "Combining speech and handwriting modalities for mathematical expression recognition," *IEEE Trans. on Human-Machine Systems*, 2017.

[6] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *CVPR*, 2015.

[7] K.-F. Chan and D.-Y. Yeung, "Mathematical expression recognition: a survey," *IJDAR*, 2000.

[8] R. Zanibbi, D. Blostein, and J. R. Cordy, "Recognizing mathematical expressions using tree transformation," *TPAMI*, 2002.

[9] F. Alvaro, J.-A. Sánchez, and J.-M. Benedí, "An integrated grammar-based approach for mathematical expression recognition," *PR*, 2016.

[10] F. Alvaro, J.-M. Benedi *et al.*, "Recognition of printed mathematical expressions using two-dimensional stochastic context-free grammars," in *ICDAR*, 2011.

[11] F. Julca-Aguilar, H. Mouchère, C. Viard-Gaudin, and N. S. Hirata, "Top-down online handwritten mathematical expression parsing with graph grammar," in *IberoAmerican Congress on PR*, 2015.

[12] H. Dai Nguyen, A. D. Le, and M. Nakagawa, "Deep neural networks for recognizing online handwritten mathematical symbols," in *ACPR*, 2015.

[13] ——, "Recognition of online handwritten math symbols using deep neural networks," *IEICE Trans. on Inform. and Sys.*, 2016.

[14] Y. Deng, A. Kanervisto, and A. M. Rush, "What you get is what you see: A visual markup decompiler," *CoRR*, 2016.

[15] Y. Deng, A. Kanervisto, J. Ling, and A. M. Rush, "Image-to-markup generation with coarse-to-fine attention," in *ICML*, 2017.

[16] J. Zhang, J. Du, and L. Dai, "Multi-scale attention with dense encoder for handwritten mathematical expression recognition," *arXiv preprint arXiv:1801.03530*, 2018.

[17] ——, "A GRU-based encoder-decoder approach with attention for online handwritten mathematical expression recognition," in *ICDAR*, 2017.

[18] J. Zhang, J. Du, S. Zhang, D. Liu, Y. Hu, J. Hu, S. Wei, and L. Dai, "Watch, attend and parse: An end-to-end neural network based approach to handwritten mathematical expression recognition," *PR*, 2017.

[19] X. Chen and C. Lawrence Zitnick, "Mind's eye: A recurrent visual representation for image caption generation," in *CVPR*, 2015.

[20] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *CVPR*, 2015.

[21] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *CVPR*, 2015.

[22] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *CVPR*, 2016.

[23] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *ICML*, 2015.

[24] J. Johnson, A. Karpathy, and L. Fei-Fei, "Densecap: Fully convolutional localization networks for dense captioning," in *CVPR*, 2016.

[25] H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu, "Video paragraph captioning using hierarchical recurrent neural networks," in *CVPR*, 2016.

[26] J. Krause, J. Johnson, R. Krishna, and L. Fei-Fei, "A hierarchical approach for generating descriptive image paragraphs," *arXiv preprint arXiv:1611.06607*, 2016.

[27] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *NC*, 1997.

[28] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[30] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR*, 2009.

[31] K. Cho, "Natural language understanding with distributed representation," *arXiv preprint arXiv:1511.07916*, 2015.

[32] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *AMACL*, 2002.

[33] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *CVPR*, 2015.

[34] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," *Text Summarization Branches Out*, 2004.