A Multilingual Dataset (MultiMWP) and Benchmark for Math Word Problem Generation

Omega Gamage, Surangika Ranathunga, Annie Lee, Xiao Sun, Aryaveer Singh, Marjana Prifti Skenduli, Mehreen Alam, Ajit Kumar Nayak, *Member, IEEE*, Haonan Gao, Barga Deori, Jingwen Ji, Qiyue Zhang, Yuchen Zeng, Muxin Tian, Yanke Mao, Endi Trico, Danja Nako, Sonila Shqezi, Sara Hoxha, Dezi Imami, Dea Doksani, Virat Kumar Pandey, Ananya Ananya, Nitisha Aggarwal, Naiyarah Hussain, Vandana Dwivedi, Rajkumari Monimala Sinha, and Dhrubajyoti Kalita

***Abstract*—We present a multi-way parallel corpus of Math Word Problems (MWPs) in nine languages, including six low-resource languages. To date, this is the largest multilingual MWP dataset available. We utilize this dataset and show the viability of using pre-trained multilingual sequence-sequence language models (prM-SLMs) for autoregressive MWP generation in both monolingual and multilingual setups, particularly for low-resource languages. We also integrate a math constraint satisfaction module with autoregressive text generation. Our extensive evaluations identify several factors that affect autoregressive text generation on prMSLMs. These include language representation in the model, model size, existence of similar languages in the model, and language script. Overall, our results reveal that autoregressive MWP generation on top of prMSLMs is very promising, even for low-resource languages.**

***Index Terms*—Multilingual dataset, multi-way parallel dataset, low-resource languages, math word problem generation, benchmark.**

Received 10 July 2023; revised 9 April 2024 and 27 February 2025; accepted 5 March 2025. Date of publication 19 March 2025; date of current version 7 May 2025. This work was supported in part by the AWS Cloud Credit for Research program, which provided GPU resources. This grant was received by Surangika Ranathunga and is acknowledged in the paper.s Acknowledgment section The associate editor coordinating the review of this article and approving it for publication was Prof. Mark Hasegawa-Johnson. *(Corresponding authors: Omega Gamage; Surangika Ranathunga.*

Omega Gamage is with the Accelr Logic, Colombo 00400, Sri Lanka (e-mail: omega.gamage@accelr.site).

Surangika Ranathunga is with the Massey University, Auckland 0632, New Zealand (e-mail: s.ranathunga@massey.ac.nz).

Annie Lee, Xiao Sun, Haonan Gao, Jingwen Ji, Qiyue Zhang, Yuchen Zeng, Muxin Tian, and Yanke Mao are with the University of Toronto, Toronto M5S 1A1, Canada.

Aryaveer Singh is with the Guru Gobind Singh Indraprastha University, New Delhi 110078, India.

Marjana Prifti Skenduli, Endi Trico, Danja Nako, Sonila Shqezi, Sara Hoxha, Dezi Imami, and Dea Doksani are with the University of New York Tirana, 1000 Tirana, Albania.

Mehreen Alam is with the National University of Computer and Emerging Sciences, Islamabad 44000, Pakistan.

Ajit Kumar Nayak is with the Siksha 'O' Anusandhan University, Bhubaneswar 751030, India.

Barga Deori is with the Dhemaji Engineering College, Scholar at NIT Silchar, Silcuri 788118, India.

Virat Kumar Pandey is with the Agra University, Agra 282004, India.

Ananya Ananya is with the Indian Institute of Technology, Bhilai 491001, India.

Nitisha Aggarwal is with the Institute of Informatics and Communication, University of Delhi, New Delhi 110021, India.

Naiyarah Hussain is with the Ambassador for CodersHQ, AI Office UAE, Dubai 00000, UAE.

Vandana Dwivedi is with the Indian Institute of Technology (BHU), Varanasi 221005, India.

Rajkumari Monimala Sinha is with the Independent Scholar, NA, India.

Dhrubajyoti Kalita is with the Cotton University, Guwahati 781001, India.

Digital Object Identifier 10.1109/TASLPRO.2025.3552936

TABLE I
EXAMPLES FROM SIMPLE AND ALGEBRAIC MWP DATASET

Type	Example
Simple	**MWP**: Bill has 9 marbles and Jim has 7 fewer marbles than Bill. How many marbles does Jim have? **Equation**: x = 9 - 7
Algebraic	**MWP**: Find three consecutive odd integers such that the sum of the first integer, twice the second integer, and three times the third is 70. **Equation**: x + 2*(x+2) + 3*(x+4) = 70

I. INTRODUCTION

MATHEMATICS is undoubtedly one of the challenging subjects for school children [1], [2]. Unlike many other subjects, Mathematics questions cannot be answered by memorizing theories and concepts. Instead, students are expected to practice by solving different types of mathematics questions [3]. However, this poses a challenge for tutors to come up with a diverse set of questions, which may be seen as an additional burden. This problem is more aggravated in the Global South, where there is an imbalanced distribution of mathematics tutors and resources [4], [5]. Despite Mathematics being a universal subject, it may be taught using the mother tongue of students (e.g., in Government schools of Sri Lanka, school children are taught mathematical concepts in either Sinhala or Tamil). This means that students cannot benefit from Mathematics problems available in other languages such as English. This is particularly true for Mathematical Word Problems (MWPs).

An MWP is expressed in natural language and expects language comprehension skills of a student in addition to mathematical skills. An MWP is formally defined as "a mathematical exercise, where significant background information on the problem is presented as text rather than in mathematical notation" [6]. Table I shows two examples of MWPs.

Natural Language Processing (NLP) based systems are continuously being experimented with to provide improved solutions in education [7]. Thus, NLP can be used to assist math

tutors in creating MWPs. Machine-assisted MWP creation in a multilingual setting can be addressed in two different ways: (1) Machine Translating MWPs written in a high-resource language such as English into the target language, and (2) machine-assisted language-specific MWP generation. A limiting factor of technique (1) is that it always depends on the availability of MWPs in a high-resource language, such as English. Moreover, the tutor should have bilingual proficiency, in order to verify the correctness of the translations.

Therefore, machine-assisted MWP generation for individual languages can be considered as the way forward, where a trained MWP generation model produces questions in the considered language. In other words, we expect the tutor to provide the starting phrase (henceforth referred to as the seed) of an MWP written in their native language, the machine will generate the MWP, and finally, the tutor checks for the correctness of the generated MWP.

However, in general, there is a great disparity in the NLP world, where high-resource languages benefit from NLP applications more than low-resource languages [8]. This plight has been already documented for NLP tasks such as Machine Translation [9]. This is mainly due to the lack of data for low-resource languages to train modern-day data-intensive NLP systems [10], and researchers not openly sharing the datasets and tool they create [11]. Thus, given that the languages used in most of the Global South are low-resource [8], data creation and publicly releasing those datasets become an important precursor for this type of NLP tasks. Therefore, this paper presents a novel multilingual MWP dataset. We started by improving the quality of an existing multilingual (English, Tamil, Sinhala) MWP dataset [12]. Then this dataset was manually translated into six more languages (Oriya, Assamese, Albanian, Urdu, Hindi, and Chinese). Each language had a main contributor, who was responsible for data creation and evaluation. This corpus (**multiMWP**) of 7470 MWPs is multi-way parallel, meaning that each MWP is present in each language. All except English, Chinese, and Hindi are low-resource languages [13]. MWPs can be in categories such as Arithmetic, Algebraic, and Geometry. The MWPs included in our corpus belong to Arithmetic (which we term 'simple' MWPs) and Algebraic categories (see Table I for an example from each of these categories). This dataset has been publicly released and is available at.[1]

Our MWP generation system is implemented by training pre-trained multilingual sequence-sequence language models (prMSLMs) for the autoregressive text generation task. We experimented with mBART50 [14], mT5 [15], M2M-100 [16], and IndicBART [17]. While Nirarepola et al. [12] also experimented with mBART50 and mT5 for autoregressive text generation, all three languages they considered were included in both models. In contrast, not all our languages are included in all the considered models. This allowed us to look into the generalization capabilities of these models for unseen languages. Our extensive evaluations show that the performance of prMSLMs for autoregressive text generation depends on the interplay of several factors: language representation in the model, model size, the existence of similar languages in the model, and language script.

Previous work that focused on autoregressive text generation suffered from errors related to math constraint violations [12], [18]. As a solution, we integrated a math constraint module [19] into our autoregressive text generation model. In addition to building language-specific MWP generation models, we built a multilingual MWP model by combining data from all the languages. This constraint-based multilingual MWP generation model is our best pick for the task.

Dataset creation and manual evaluation of results were inspired by the concept of participatory research in language data creation [20]. In participatory NLP research, rather than employing paid workers for data creation and model evaluation, NLP researchers and language experts (as well as language users) contribute to data creation, and become co-authors of the publication. We followed a hybrid model, where the main language contributors were given the option to either pay their language workers, or offer them co-authorship. In summary, this paper makes the following contributions:

- A multi-way parallel dataset consisting of MWPs (nine languages, with six being low-resource) for two domains. Thus, our dataset has 18 separate datasets.
- A detailed analysis of the autoregressive text generation capabilities of prMSLMs, with a special focus on low-resource languages
- A constraint-based MWP generation model

II. Related Work

A. MWP Datasets

Table II summarises the currently available MWP datasets. Here, the MAWPS dataset [21] is an extension of [22], [23], [24], [25]. Niyarepola et al.'s [12] dataset extends Liyanage and Ranathunga's [18] dataset and the Dolphin18 [26] dataset. Math23k [27] dataset was originally compiled for Chinese and Wang et al. [19] subsequently translated this dataset to English. Liyanage and Ranathunga [18] and Niyarepola et al. [12] are the only multilingual datasets, and only Niyarepola et al.'s [12] dataset is multi-way parallel. However, both these datasets contain MWPs for English, Sinhala, and Tamil only. In summary, there is only a handful of multilingual MWP datasets. The maximum number of languages covered in a dataset is three.

B. MWP Generation

MWP generation techniques can be categorized as question re-writing [32], template-based generation [33], [34], [35], [36], and generation with Neural Networks (NNs). NN-based techniques can be further categorized as those that use prMSLMs [12], [19] and those that use architectures such as Recurrent Neural Networks (RNNs) [6], [18], [37] and Variational Auto-Encoders (VAEs) [38], [39].

NN-based techniques generate MWPs either by considering an equation and a context [6], [19], [38], [39], [40], or in an autoregressive manner [12], [18], [37]. As far as we know, these autoregressive techniques are the only models that were tested

[1]https://github.com/OmegaGamage/multiMWP/tree/master/dataset

TABLE II
DATASETS FOR MATHEMATICAL WORD PROBLEMS (MWPS)

Dataset Reference	Number of MWPs	Problem Type
[28]	1024	Linear Algebra (single and two-equation problems)
[22]	395	Arithmetic (addition and subtraction for third to fifth grade level)
SINGLEEQ [23]	508	Algebraic (single equations with multiple operations)
[24]	1404	Arithmetic (involving addition, subtraction, division, and multiplication)
allArith [25]	600	Arithmetic (comprising two operations)
MAWPS [21]	3734	Arithmetic and Algebraic problems
DRAW-1k [29]	1000	Algebraic problems
[30]	1878	Numerical Word problems
dolphin18k [26]	18460	MWPs at primary and secondary school level
MathQA [31]	37259	Problems across domains such as physics, probability, counting, geometry, and numerical computations
Math23k [27]	23162	Linear Algebra (at elementary level)
[18]	4350	Arithmetic (simple) and Algebraic problems
[12]	7370	Arithmetic (simple) and Algebraic problems

TABLE III
LANGUAGE DETAILS (DATA ON THE LANGUAGE COVERAGE OF MODELS WAS EXTRACTED FROM [14], [15], [16], [17])

Language	ISO 639-3	Family	Branch	Script	mBART50	IndicBART	mT5			M2M-100
					#Sentences(M)	#Sentences(M)	Tokens(B)	Pages(M)	mT5(%)	Coverage
Albanian	sqi	Indo-European	Albanian	Latin	0	0	4	4.1	0.76	Yes
Assamese	asm	Indo-European	Indo-Iranian	Bengali	0	1.4	0	0	0	No
Chinese	zho	Sino-Tibetan	Sinitic	Han	10.08	0	39	55	1.67	Yes
English	eng	Indo-European	Germanic	Latin	12.61	54.3	2,733	3,067	5.67	Yes
Hindi	hin	Indo-European	Indo-Iranian	Devanagari	1.33	63.1	24	19	1.21	Yes
Oriya	ory	Indo-European	Indo-Iranian	Oriya	0	7	0	0	0	Yes
Sinhala	sin	Indo-European	Indo-Iranian	Sinhala	0.57	0	0.8	0.5	0.41	Yes
Tamil	tam	Dravidian	Southern	Tamil	0.61	31.5	3.4	3.5	0.73	Yes
Urdu	urd	Indo-European	Indo-Iranian	Perso-Arabic	0.03	0	2.4	1.9	0.61	Yes

TABLE IV
ERRORS THAT WERE IDENTIFIED DURING DATA CLEANING

Error Type	Description	Example
Grammar error	Sentence violates grammar rules of the language	Supun wrote essay and he used 5 pens and 4 pencils. How much more pens than pencils did Supun use? Here essay should have the article "an"
Co-reference	Inconsistent co-reference	**Nimal** ran 8 mile and walked 5 mile. How much farther did **Vimal** run than walk? Here, the second sentence has the proper noun Vimal, instead of Nimal
Random characters	Non-alphanumeric characters can occur within sentence/word structure	අබණ්ඩව පුරිණ සංඛ්‍යා තුනක ප්‍රතිඵලය 54 වේ, පුරිණ සංඛ්‍යා තුනම සොයා ගන්න. Here, ප්‍රතිඵලය should be change to එකතුව
Transliteration error-1	Inclusion of source language script in target language sentence	ඩාර්නල් වට 0.875ක් දිව ගිය අතර පසුව වට 0.75ක් දුවමින් විවේකයක් ගත්තේය. Darnel මුළු වට කීයක් කළාද? Here, "Darnel" is an English word(noun), instead of it ,ඩාර්නල් should be used
Transliteration error-2	Source language words are represented in the script of the target language rather than being translated	ටෝයි ප්ලේන් මීටර් 3 ක් පැහා මීටර් 2 ක් පියාසර කළේය. ටෝයි ප්ලේන් පියාසර කරනවාට වඩා කොපමණ පැහා ගියාද? Here ටෝයි ප්ලේන් is an incorrect translation and it is the English word "Toy plane" written in Sinhala scripts, correct translation is, සෙල්ලම් ගුවන්යානයක්
Errors due to translation gap	Translation gap refers to the challenge of achieving equivalent meaning in the target language due to disparities between the source and target languages	මෝගන් ඇගේ දෙවන SAT උත්සාහයේ 10% ක දියුණුවක් ගැන බබළෙවීත් සතුටු විය. අය දෙවන උත්සාහයේදී ලකුණු 1100ක් ලබා ගත්තාය. ඇගේ පළමු ලකුණු මොනවාද? Here, "SAT" is an exam in the USA and, the Sinhala language does not have a proper term for it
Partial translation	Only part of the text is translated. Resulted text is incoherent	42 கி. ஒரு நில்லையத்துடன் மற்றமொரு குன்று குணம் பெரியது. எம் நிலில தேடுங்கள். Complete translation should be: இரண்டு நேர்மறை முழு எண்களுக்கு இடையிலான வேறுபாடு 42 ஆகும். ஒரு முழு எண் மற்றதை விட குன்று மடங்கு பெரியது. முழு எண்களைக் கண்டறியவும்.
Source language errors	Mistake in the source language text that can make it difficult to accurately translate the text into the target language.	অনিলে পাড়ী কিনিছিল আৰু তেওঁ ২০ খন নগদ ধনৰ অভাৱ আৰু ১০ খনত চকৰ অভাৱ আছিল। চকতকৈ কিমান বেছি নগদ ধন অনিলে ব্যৱহাৰ কৰিছিল? Here, অভাৱ means "lacks". In the corresponding English MWP, lacks was mistakenly typed instead of "lakhs". This mistake had confused several translators in few other languages. Correct Assamese term is "লাখ টকা"
Lexical errors	Uses the wrong word in a translation	ھیں؟ سنگترہ کتنے پاس کے ومل ھیں۔ سنگترہ کم 6 سے لاسل پاس کے ومل اور ھیں سنگترہ 10 میں لاسل Here سنترے (oranges) should be used instead of سنگترہ (tangerine)
Typos	Errors due to typing mistakes	නාලින්ට කිරිගරු 11 11 ක් සහ සීමට නාලින්ට වඩා කිරිගරු 9 9 ක් ඇත. සීමට කිරිගරු කීයක් තිබෙද? Here repetition of 11 and 9 are typos

in a multilingual setting. Niyarepola et al. [12] showed that autoregressive text generation with prMSLMs significantly outperforms their RNN counterpart, when the considered languages are included in prMSLM pre-training.

III. THE MULTILINGUAL MWP DATASET

We extended the multi-way parallel dataset presented by Niyarepola et al. [12] to six more different languages and present the multiMWP dataset. The statistics of the languages can be found in Table III. Language selection was purely based on the availability of an NLP researcher to provide the data.

A. Data Cleaning

Niyarepola et al. [12] carried out a manual evaluation of their dataset and identified some quality issues. However, no subsequent effort was taken to clean the dataset. Therefore, the first author manually cleaned both the English and Sinhala MWP datasets of Niyarepola et al. [12] (English and Sinhala datasets were selected because these are the two languages that the first two authors are familiar with). Through this process, a variety of errors including grammatical, typographical, and lexical inaccuracies were identified and subsequently corrected (see Table IV). Any misalignment in the Sinhala-English parallel corpus was also fixed.

B. Manual Translation

For each language, an NLP researcher, who was a native speaker of the language, acted as the main language contributor, and was responsible for providing the translated data for that language. The NLP researcher and/or some translators they

TABLE V
DATASET STATISTICS

Lang.	Context	Num. Words	Unique Words	Words/Sent.	Chars/Sent.
Albanian		57783	5783 (10.01%)	18.29	84.1
Assamese		54894	4870 (8.87%)	17.37	90.07
Chinese		69216	4099 (5.92%)	21.9	36.09
English		62229	4043 (6.5%)	19.69	87.77
Hindi	Simple	65624	3586 (5.46%)	20.77	83.29
Odia		61415	4815 (7.84%)	19.44	93.26
Sinhala		58211	5330 (9.16%)	18.42	92.97
Tamil		45525	7034 (15.45%)	14.41	114.95
Urdu		68314	3497 (5.12%)	21.62	80.26
Albanian		87924	2872 (3.27%)	20.88	93.05
Assamese		56108	2258 (4.02%)	13.33	77.05
Chinese		86390	1611 (1.86%)	20.52	32.01
English		79325	1638 (2.06%)	18.84	84.79
Hindi	Algebraic	74966	1505 (2.01%)	17.81	74.37
Odia		68197	1813 (2.66%)	16.2	81.94
Sinhala		60724	2762 (4.55%)	14.42	76.53
Tamil		56163	3455 (6.15%)	13.34	97.82
Urdu		76902	1446 (1.88%)	18.27	67.43

found, carried out the translations. Each translator was either paid or made a co-author. All the translators were given the following set of instructions to follow during translation.

1) Keep the same question order as the English version
2) Keep the same number of sentences as in the English MWP
3) Maintain the syntactic structure of the English sentence in the target language sentence, if the target language syntax allows it (e.g., if the English MWP is in active voice, the target language MWP should be kept in active voice)

On average, there are two sentences per question, with a maximum of four sentences. A detailed analysis of the dataset statistics is included in Table V.

C. Quality Estimation of the Dataset

The quality of the translated datasets was estimated automatically as well as manually.

1) Automatic Quality Estimation: Our automatic quality estimation method is based on sentence similarity. The intuition is, if the translation of an MWP is correct, then it should have a high similarity with the corresponding English MWP.[2] To measure the similarity between sentence pairs, we utilized the cosine similarity of MWP embeddings. We considered LaBSE [41] and XLM-R [42] to generate MWP embeddings.

First, we plotted the similarity scores for each English-Sinhala sentence pair in the dataset using both LaBSE and XLM-R embeddings. A thorough analysis of the similarity scores in relation to the perceived quality of the English-Sinhala translations indicated a stronger association with human assessment of translation quality in the XLM-R embedding similarity scores. Therefore, we chose XLM-R sentence embeddings for the similarity measure.

This analysis helped us to identify a potential threshold (0.4) of the cosine similarity score to distinguish good translations from bad ones in Sinhala-English translations. Then this threshold was used to identify MWPs that may have been of low quality in other languages. Specifically, this approach and a plot visualization facilitated the identification of the following translation errors:

- English MWP has been copied to the target side (indicated by a perfect similarity score of 1)
- Discrepancies in the ordering of MWPs compared to that of the English dataset (evidenced by sudden drops in similarity scores throughout a range in the plot)
- Duplicate MWPs on the target side (indicated by equal similarity scores)
- Some MWPs have not been translated into the target language (identified through simple error checks).

The identified errors were marked, and the dataset was sent back to the contributor to manually fix.

2) Manual Quality Estimation: In order to verify the quality of the manual translation, we used the Direct Assessment method [43], which is commonly used in translation evaluation research. We selected three bilingual speakers for each language pair. Each evaluator was assigned 200 translated MWPs along with the original English MWP. They were asked to rate the translated version with respect to adequacy and fluency and give a rating between 1–100 as per the criterion defined in [43], where 0–10: incorrect translation, 11–29: a translation with few correct keywords, but the overall meaning is different from the source, 30–50: a translation with major mistakes, 51–69: a translation which is understandable and conveys the overall meaning of the source but contains typos or grammatical errors, 70–90: a translation that closely preserves the semantics of the source sentence and 91–100: a perfect translation.

The evaluation results, displayed in Table VI, demonstrate the effectiveness of the translation process across various languages and contexts. Specifically, for Albanian, Assamese, Odia, and Sinhala, over 80% of the MWPs have a translation score exceeding 70. For Chinese and Hindi, the corresponding figure surpasses 75%. However, Tamil is an exception, with only 64%

TABLE VI
PERCENTAGE DISTRIBUTION OF DIRECT ASSESSMENT SCORES FOR TRANSLATED SENTENCES ACROSS LANGUAGES AND CONTEXTS

Dataset		Rating					
Language	Context	0-10	11-29	30-50	51-69	70-90	91-100
Albanian	Simple	1.3	1.7	1.7	4.3	12	79
	Algebraic	0.3	3	1.7	12	12	71
Assamese	Simple	0.5	0	0.5	10	12.5	76.5
	Algebraic	9.5	2	3	4	25.5	55
Chinese	Simple	1.3	3	5.3	9	31	50.3
	Algebraic	2.7	2.7	11	5	25	53.7
Hindi	Simple	0.7	0.7	5.3	13.7	10.3	68.3
	Algebraic	0.7	1.7	6.3	9.3	6	75.7
Odia	Simple	0	0	0	2.7	15.3	82
	Algebraic	0	0	0.7	0.7	10.3	87.7
Sinhala	Simple	1	1	1	3	6	88
	Algebraic	2	4	6	6	10	72
Tamil	Simple	1.7	4	12.3	18	44.3	19.7
	Algebraic	11	6.3	9	18	21.3	34.3
Urdu	Simple	0	0	1.7	9	13.3	76
	Algebraic	0.3	0	4.3	8.3	23.3	63.7

[2]In our study, we found that exact translations often do not achieve a similarity score of 1.0 due to natural variance across different languages. High-quality translations typically receive scores between 0.95 and 0.98. A score of 1.0 indicates that the English text was copied directly to the target language, which is a type of error we identified using this method.

and 55.6% of MWPs in the Simple and Algebraic contexts (respectively), reaching scores above 70.

IV. Methodology

A. Autoregressive Text Generation

Previous work has shown that autoregressive text generation, either with RNNs or prMSLMs, is a viable option for MWP generation [12], [18]. In particular, Niyarepola et al. [12] showed that autoregressive text generation, even with a small seed, is effective for low-resource languages such as Sinhala and Tamil. On the other hand, we did not have any contextual information related to the MWPs in the considered languages in order to employ NN-based techniques that generate MWPs by considering an equation and a context [6], [19], [38], [39] (Note that we would need contextual information in nine different languages).

Therefore, we resort to autoregressive text generation. Our dataset is not large enough to train an autoregressive model from scratch. Therefore, we used prMSLMs to initialize our text generation model. To be specific, we used the conditional generator option in prMSLMs. Input to the model is the seed text (the starting portion) of the MWP, and the model is expected to generate the rest. For all the experiments, we selected the first 25% of the words in the original MWP as the seed. Note that this is the smallest seed size that was considered by Niyarepola et al. [12]. Via a human evaluation, they have shown that text generation using this seed size is still more effective than a tutor generating a question from scratch.

B. Constraint-Based Text Generation

Generating MWPs presents unique challenges as it requires the generation of linguistically coherent and grammatically correct text while also ensuring compliance with mathematical rules. While traditional natural language generation (NLG) approaches have been successful in addressing linguistic constraints, the satisfaction of mathematical constraints in MWPs has received less attention. A significant portion of the errors in the model proposed by Niyarepola et al. [12] was attributed to the basic autoregressive generation on prMSLM models failing to adhere to these mathematical constraints. For instance, in the MWP shown in Fig. 1, if Anil's age is 7, it results in an unrealistic age of -3 years for Harin.

The only NN-based work that we could find on constraint-based MWP generation is Wang et al.'s [19] model. They utilized an equation consistency constraint for mathematical equations to improve the mathematical validity of the generated MWP. Their model consists of two sub-models:

1) MWP generation model
2) MWP to equation conversion model (mwp2eq model)

The overarching objective for the MWP generation model, represented by p_Θ, is illustrated in Equation (1).

$$\mathcal{L} = \mathcal{L}_{\text{mwp}} + \alpha \mathcal{L}_{\text{eq}} \tag{1}$$

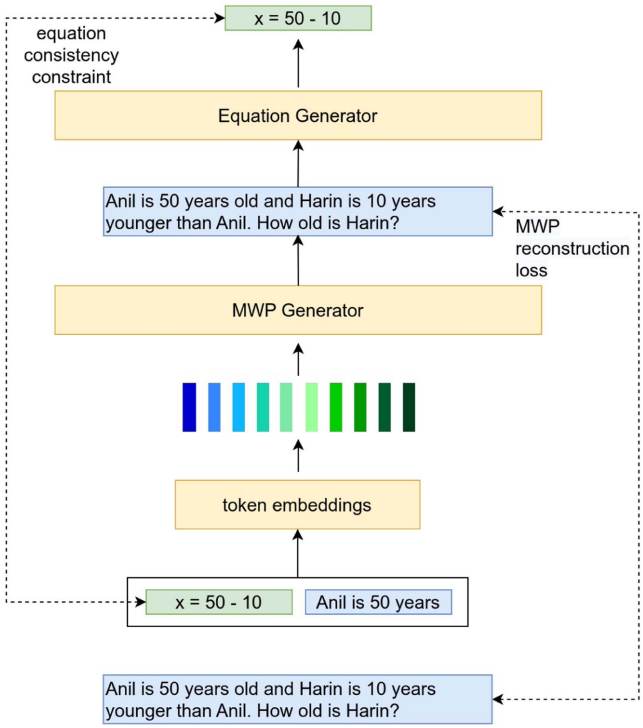

Fig. 1. Constraint-based MWP generation model (Adapted from [19]).

Where $\mathcal{L}$ denotes the total loss, $\mathcal{L}_{\text{mwp}}$ the MWP loss, $\mathcal{L}_{\text{eq}}$ the equation consistency loss, and α the hyperparameter balancing the constraint term.

Here $\mathcal{L}_{mwp}$ is the negative log-likelihood objective for MWP generation.

$$\mathcal{L}_{\text{mwp}} = \sum_{t=1}^{T} -\log p_\Theta(m_t | E, \mathbf{s}, \{m_r\}_{r=1}^{t-1}) \tag{2}$$

The notations $\{m_r\}_{r=1}^{t-1}$, E, and $\mathbf{s}$ in (2) denote the MWP as a sequence of tokens, equation, and seed, respectively.

We further introduce an equation consistency constraint, denoted by $\mathcal{L}eq$, which is defined as:

$$\mathcal{L}_{eq} = \sum_{t=1}^{T} -\log p_\Phi(e_t | M', e_1, \ldots, e_{t-1}) \tag{3}$$

In (3), p_Φ represents the mwp2eq generation model, M' symbolizes the generated MWP, and e_t refers to a sequence of mathematical symbols. This approach models the problem as a sequence-to-sequence (seq2seq) task, where the input is an MWP and the output is its corresponding equation, represented as a sequence of mathematical symbols e_t.

We modified Wang et al.'s [19] model to suit our research objectives. Specifically, we omit the keyword selection model, which requires contextual keywords such as nouns and proper nouns for effective performance. Extracting such linguistic information from MWPs, particularly in a multilingual dataset with low-resource languages, poses challenges. As an alternative, we used a seed of 25% of the words. Additionally, we replaced their decoder-based GPT-2 with a prMSLM. The main reasoning behind this design choice is our seed-based generation

TABLE VII
RESULTS FROM BASELINE MODELS INCLUDING GPT-4

Context	Language	mBART50		IndicBART		mT5		M2M-100		GPT-4(gpt-4-0613)	
		BLEU-4	METEOR	BLEU-4	METEOR	BLEU-4	METEOR	BLEU-4	METEOR	BLEU-4	METEOR
Simple	Albanian	*45.38*	0.6385	30.56	0.5585	**46.00**	*0.6431*	45.33	**0.6442**	5.69	0.3187
	Assamese	**48.74**	**0.6371**	*46.59*	*0.6220*	46.31	0.6162	15.22	0.4221	7.05	0.2889
	Chinese	*48.89*	*0.4787*	5.00	0.1720	**50.52**	**0.4850**	48.83	0.4767	8.76	0.3093
	English	*52.67*	*0.7202*	51.87	0.7102	**53.12**	**0.7214**	51.27	0.7141	8.09	0.4511
	Hindi	*50.47*	*0.6450*	49.4	0.6312	48.5	0.6312	**50.6**	**0.6480**	10.37	0.3528
	Oriya	**44.76**	**0.6514**	43.46	0.6391	41.12	0.6189	*44.21*	*0.6506*	8.63	0.2713
	Sinhala	*44.47*	*0.6204*	0.96	0.1454	41.57	0.5922	**46.01**	**0.6388**	6.28	0.268
	Tamil	**41.34**	**0.6145**	39.84	0.6025	39.05	0.5953	*40.04*	*0.6075*	5.3	0.3158
	Urdu	**51.43**	**0.6383**	40.11	0.5409	48.52	0.6123	*50.89*	*0.6365*	11.62	0.3608
Algebraic	Albanian	41.50	0.5447	38.49	0.5269	*41.64*	**0.5524**	**41.81**	*0.5470*	20.9	0.3199
	Assamese	**36.18**	**0.4996**	*33.91*	*0.4853*	33.06	0.4766	11.33	0.3236	9.21	0.2391
	Chinese	34.74	0.2024	7.78	0.1337	**35.57**	**0.2096**	*35.02*	*0.2032*	21.96	0.3568
	English	**46.98**	**0.6521**	42.97	0.6239	45.38	0.6427	*46.61*	*0.6450*	22.62	0.4507
	Hindi	*44.41*	*0.5742*	42.91	0.5651	43.34	0.5711	**45.49**	**0.5841**	17.04	0.3399
	Oriya	**41.01**	**0.5947**	36.35	0.5453	38.85	0.5708	*40.79*	*0.5902*	11.61	0.2605
	Sinhala	**21.13**	**0.4253**	0.67	0.1004	18.1	0.408	*20.74*	*0.4200*	6.15	0.2226
	Tamil	*27.47*	*0.4467*	**27.58**	**0.4670**	26.96	0.4595	26.82	0.4459	7.4	0.2421
	Urdu	**46.21**	**0.5641**	37.82	0.4923	43.62	0.5486	*44.76*	*0.5603*	14.03	0.2978

Bold font denotes best results, *italic font* denotes second best. Grayed out results indicate that the language is not covered in the model.

approach, which facilitates a tutor to create an MWP without providing any contextual information.

The training process consists of two phases. In the initial stage, we trained the 'mwp2eq' model independently using the MWP dataset. The 'mwp2eq' model is a sequence-to-sequence (seq2seq) model designed to generate the underlying equation, which serves as the mathematical constraint, for a given MWP. It was adapted from Wang et al. [19], which provides more details on its design and implementation.

The subsequent stage involves the joint training of both the MWP generation and 'mwp2eq' models. The purpose of this stage is to fine-tune the MWP generation model by leveraging feedback from the 'mwp2eq' model, thereby enhancing the mathematical validity of the generated MWPs. The interplay between the MWP generation model and the mwp2eq model during this joint training phase is depicted in Fig. 1.

C. Multilingual Text Generation

For multilingual text generation, we reused the baseline (Section IV-A) and the constraint-based text generation (Section IV-B) models. However, we trained one model for all the languages. Here, the training was done in one go, and the dataset was a concatenation of MWPs from all the languages. The training set was created by randomly integrating MWPs from the training subset of each language.

V. EXPERIMENT SETUP

The model selection was based on the coverage of the languages in our dataset. We selected four models, namely mBART50, mT5, M2M-100, and IndicBART. IndicBART was chosen due to its compactness and relevance to the Indic languages present in our dataset. The pre-trained models were obtained from Huggingface.[3] Tables III and VIII carry more information about the models and their language support.

TABLE VIII
SUMMARY OF MODELS

Model	Huggingface model	#Parameters	#Languages
mBART50	mbart-large-50	610M	50
IndicBART	indic-bartSS	244M	12
mT5	mt5-base	582M	101
M2M-100	m2m100_418M	484M	100

All experiments were conducted using an A6000 GPU with 48 GB of memory. To train models for various languages, a consistent approach was adopted by maintaining a fixed number of epochs (30) and batch size (8) while manipulating other relevant hyper-parameters related to training. Values of some other hyper-parameters are given in Table XV, in Appendix VII. Additionally, an early stopping mechanism was implemented to prevent over-fitting. An issue encountered during the experimentation process was that when training models for each language individually, it was necessary to adjust the learning rate and the number of warm-up steps to obtain better results. It was observed that the optimal hyper-parameters for one language may not necessarily produce favorable results for another language. For all the experiments, a train:validation:test split of 60:10:30 was used.

VI. RESULTS AND EVALUATION

In this study, both human evaluation and automatic metrics were used for evaluation. The two metrics used were BLEU-4 [44], and METEOR [45]. All metrics were calculated using the Huggingface library.[4]

A. Experiments on Baseline Models

In this study, we evaluated the performance of language models after fine-tuning them on the MWP datasets. We refer to

[3][Online]. Available: https://Huggingface.co/

[4][Online]. Available: https://huggingface.co/

TABLE IX
MBART50 RESULTS FROM ALL APPROACHES USING THE SIMPLE DATASET

Language	Baseline		Constraint-based		Baseline + Multilingual		Constraint-based + Multilingual		Translation	
	BLEU-4	METEOR	BLEU-4	METEOR	BLEU-4	METEOR	BLEU-4	METEOR	BLEU-4	METEOR
Albanian	**45.38**	*0.6385*	43.12	0.6539	38.16	0.5747	*45.22*	**0.6645**	13.57	0.3317
Assamese	48.74	0.6371	**50.05**	**0.6772**	37.26	0.5318	*49.31*	*0.6698*	9.83	0.2922
Chinese	48.89	0.4787	*51.34*	**0.6852**	48.99	0.63	**52.45**	*0.6836*	16.24	0.3627
English	*52.67*	0.7202	**52.91**	**0.7572**	48.94	0.6974	51.5	*0.7369*	-	-
Hindi	50.47	0.6450	**51.45**	**0.6839**	47.31	0.6235	*50.89*	*0.6775*	25.18	0.4535
Oriya	44.76	0.6514	**45.02**	**0.6861**	33.06	0.5437	*44.96*	*0.6826*	7.25	0.2291
Sinhala	*44.47*	0.6204	44.16	*0.6451*	38.71	0.579	**44.74**	**0.6498**	3.11	0.1638
Tamil	*41.34*	0.6145	**42.97**	**0.6691**	36.51	0.5808	40.8	0.6445	10.83	0.3463
Urdu	*51.43*	0.6383	**52.4**	**0.6815**	47.61	0.6121	*51.43*	*0.6696*	25.25	0.4576

Bold font denotes best results, *italic font* denotes second best.Grayed out results indicate that the language is not covered in the model.

these fine-tuned models as our baselines. The baseline results of different models are shown in Table VII. Similar to Niyarepola et al.'s [12] observations, the mBART50 model performed better compared to other models. This could be due to it being relatively larger than others (see Table VIII). In particular, for simple MWPs, the mBART50 model generated the highest BLEU-4 and METEOR scores for four languages. Additionally, it obtained the second-highest BLEU-4 scores for all other languages. In more detail, mBART50 performed the best in English, followed by Hindi and Chinese. We believe this is due to the high representation of these languages in mBART50. A surprising result is Urdu, which is relatively under-represented. Urdu's high scores might be due to its syntactic similarity to Hindi and script similarity to Arabic. The high result of Albanian, which is missing in mBART50, might be due to its script being Roman. Lee at al. [46] made a similar observation for Afrikaans in Neural Machine Translation. Similarly, Oriya and Assamese languages seem to benefit from related languages such as Hindi and Bengali. Hu et al. [47] noted a similar behavior for Indo-European languages in encoder-based models such as mBERT, and Ranathunga and de Silva [8] hypothesized that this is due to this language family being dominant in prMSLMs. Similar cross-lingual generalization has been observed in other multilingual models. Prior work [48] found that models can support additional languages beyond those explicitly present in their pretraining corpus, likely due to vocabulary overlap and linguistic similarities. The scores for Tamil are even below the scores achieved by Albanian, Assamese, and Oriya, which are not represented in the mBART50 training data Table III. This discrepancy can be ascribed to several factors. Primarily, the insufficient representation of Tamil and associated Dravidian languages within the mBART50 model could have contributed to this outcome. Moreover, as illustrated in Table VI, the relatively low quality of the Tamil dataset might have also had an impact. The results from Algebraic MWPs exhibited similar trends, albeit with relatively lower scores.

mT5 performed the best for Chinese, English, and Albanian for Simple MWPs, and only for Chinese for Algebraic MWPs. However, the results of Oriya and Assamese show its generalization capabilities to unseen languages, similar to mBART50.

M2M-100 also shows on-par results for languages already represented in it. It is the best for Hindi and Sinhala simple MWPs, as well as for Albanian and Hindi Algebraic MWPs.

TABLE X
INDICBART RESULTS: BASELINE AND ROMANIZED DATASET FOR SINHALA AND TAMIL

Context	Language	Baseline		Romanized	
		BLEU-4	METEOR	BLEU-4	METEOR
Simple	Sinhala	0.96	0.1454	26.98	0.3797
	Tamil	39.84	0.6025	19.52	0.3398
Algebraic	Sinhala	0.67	0.1004	21.5	0.3561
	Tamil	27.58	0.467	21.46	0.3431

The results of IndicBART are rather disappointing. Most of the time, it does not outperform other models even for Indic languages, despite being specifically trained on those languages with larger dataset sizes. These observations are in alignment with those made by Dabre et al. [17] for text summarization and question the practicality of utilizing language-family-specific small multilingual models.

Generalization capability of IndicBART to unseen languages is also not consistent across languages. Out of the languages missing in IndicBART, results for Urdu and Albanian are generally good. This could be due to Urdu having a high syntactic similarity to Hindi and Albanian using the same script as English. On the other hand, Sinhala and Chinese have very low results, which might be due to script differences. In order to further verify the impact of the script, we converted Sinhala and Tamil MWPs to the Roman script using a transliteration tool [49].[5] In Table X, we see a significant gain for Sinhala. On the other hand, the results for Tamil show a decrease when comparing the baseline and romanized results.

Table VII indicates notable variations in prMSLMs performance between high-resource and low-resource languages, as well as between Simple and Algebraic contexts.

In terms of model performance, mBART50 demonstrated the most consistent results, exhibiting the lowest standard deviation in BLEU-4 metrics across both Simple (3.77) and Algebraic (8.86) contexts. This was followed by mT5 (4.68, 9.13), M2M-100 (11.27, 12.56), and IndicBART (18.77, 15.30). The data further suggests that language representation in the model is a critical determinant of their performance. This is evident from the performance of English, the language with the most

[5]Earlier we converted the Tamil script to Devanagiri, using IndicBART utilities.

TABLE XI
IDENTIFIED ERRORS IN THE GENERATED MWPS

Error Type	Description	Examples
Co-reference issues	Inconsistent co-reference	White is 19 years old and Black is 7 years younger than white. How old is Kalu? Here the second sentence has the proper noun Kalu, instead of Black
Grammatical errors	Violates grammar rules of the language	Emiley ran 14 mile and walked 15 mile. How much farther did Emiley walk than run? Here mile, should be plural;"miles"
Misspellings	Incorrectly spelled word/s	ලසල්ට පැකැට් 12ක් ඇති අතර රෝසිට් ලසල්ට වඩා 6ක් අඩුවෙන් පැකැට් ඇත.රෝසිට් පැකැට් කොපනම ප්‍රමාණයක් තිබෙද්? Here, correct spellings for "කොපනම" is කොපමණ
Incomplete sentences	Incomplete sentences are present in the MWP	පූසෙකු සැතපුම 2 ක් ගමන් කල අතර සැතපුම් පසුන් පසු කල මුළු දුර කොපමණද? Possible correction: පූසෙකු සැතපුම 2 ක් ඇවිද්ද අතර සැතපුම් 5 ක් දිවිවා. පූසා ගිය මුළු දුර කොපමණද?
Unsolvable problems	Not enough information or contradictions	Daren mblodhi 18 dardha nga pema e dardhës. Sa dardha kishin mbetur? English translation: Daren picked 18 pears from the pear tree. How many pears were left? Cannot solve without knowing the initial number of pears in the tree.
Unrealistic	Solvable, but solution is not realistic	نرمل نے 8 کتابیں خریدیں اور 9 کو غل دیں۔ کتابیں کتنی اب پاس کے نرمل ہیں؟ English translation: Nirmal bought 8 books and gave 9 books to Nimal. How many books does Nirmal have now? Answer is -1
Trivial problems	MWP is trivial and easy to solve	170 வௌவால்கள் உள்ளன. வௌவால்களின் மொத்த எண்ணிக்கை என்ன? English translation: There are 170 bats. What is the total number of bats? Asking about total number of bats. But the answer is already given in the question itself
Unit issues	An inconsistent unit is linked to a numerical value	奈良 有 12 把 钥匙 , 哈里 比 奈良 多 6 把 钥匙 . 哈里 有 几个 钥匙 ? The unit of the question should be"把" instead of"个"

representation in all model (except for IndicBART, where Hindi is more represented), which consistently achieves the highest BLEU scores.

Language-wise, all the models typically exhibit higher standard deviation for low-resource languages compared to high-resource languages, a trend that is expected based on the disparity in available training data. However, an exception was observed with IndicBART. This model, primarily trained on low-resource Indic languages and not on high-resource languages like Chinese, deviated from this trend (except for Hindi).

In terms of datasets, the Simple dataset demonstrates a lower standard deviation compared to the Algebraic dataset, indicating more uniform performance across different languages and models.

In summary, we see that factors such as language representation, the existence of similar languages in the pre-trained model, and language script have a noticeable impact on model performance for autoregressive text generation.

TABLE XII
PERCENTAGES OF DIFFERENT TYPES OF ERRORS FOUND IN SIMPLE MWPS GENERATED BY THE BASELINE AND THE CONSTRAINT-BASED TEXT GENERATION MODELS

Language	Context	Co-ref.	Spell.	Unit	Gram.	Math.	Incomplete
Albanian	Baseline	**3**	**11**	6.3	7.7	**11.3**	**8.7**
	Constraint-based	3.3	11	**5.7**	**6.3**	13	9.3
Chinese	Baseline	16.3	1.3	3.3	4.7	15.6	6.3
	Constraint-based	**13.3**	**0**	**1.7**	**3.7**	**11.6**	**5.7**
English	Baseline	7	1.3	**0**	13	16.1	**2.3**
	Constraint-based	**3.7**	**1**	0.3	**11.7**	**11**	3
Hindi	Baseline	14	5.3	1	15.7	14	3.7
	Constraint-based	**13.3**	**4.3**	**0.7**	**11.3**	**10.4**	**3.3**
Odia	Baseline	18.7	3.3	**0**	15.3	9.3	**2**
	Constraint-based	**11.3**	**1.7**	2	**7.7**	**4**	4
Sinhala	Baseline	8.3	**18.3**	2.3	8.3	**17**	**4.3**
	Constraint-based	**7.3**	22.7	**1.3**	**8**	20.7	**4.3**
Tamil	Baseline	15.3	11	3.7	**36.3**	23.03	8.3
	Constraint-based	**8.3**	**7.3**	**1.3**	36.7	**17.6**	**7**
Urdu	Baseline	9.7	12.3	32.3	9.3	17.3	**3**
	Constraint-based	**8**	**11.7**	**29.3**	**2.7**	**14.4**	12

Bold font denotes lowest errors.

TABLE XIII
A SAMPLE MWP GENERATED BY BASELINE AND THE CONSTRAINT-BASED MODELS

Model	Generated MWP	Ans:
Baseline	Tom got 3 eggs but 4 were actually damaged. How many good eggs did Tom get?	-1
Constraint-based	Tom got 3 eggs but 2 were actually damaged. How many good eggs did Tom get?	1

B. Improvements to the Baseline Model

Since mBART50 showed the overall best performance, we used it for further experiments. Table IX summarizes the results of the improvements we carried out on the mBART50 baseline system. To be specific, we trained the baseline model in a multilingual manner, integrated the constraint-satisfaction model with the baseline, and finally trained the combined model in a multilingual manner. Results are reported only for the simple MWP dataset, as it is the only dataset that contains equations that are compatible with Wang et al.'s [19] approach.

As per the METEOR scores, the constraint-based model outperforms the baseline consistently. The same was observed for BLEU-4, except for the Albanian and Sinhala. This improvement can be attributed to the use of equation consistency constraint during the training process of the constraint-based model, which improves the mathematical validity of the generated MWPs.

The multilingual version of the baseline lags behind the per-language model in all languages except for Chinese. The average result (BLEU-4) drop for languages included in mBART50 is 3.53, and the same for languages not included in mBART50 is 10.13. However, when the constraint-based model is implemented in a multilingual manner, it outperforms the baseline for all the languages with respect to METEOR score. The same is observed with respect to BLEU-4, except for Albanian, English, and Tamil. This model even outperforms the constraint-based per-language model for Chinese and Sinhala (and Albanian with respect to METEOR score). Considering the added convenience of a single model that can support multiple languages, our best pick is the constraint-based model implemented in a multilingual manner.

TABLE XIV
COMPARISON OF INADEQUATE OUTPUTS GENERATED BY THE BASELINE VERSUS CONSTRAINT-BASED MODELS

Input Seed	Baseline Model Output	Constraint-Based Model Output	Expected Output
How many plants should	How many plants should the park have?	How many plants should be planted in 15 days?	How many plants should be planted per column if you have 52 plants and 15 columns in all?
Marc sold 563 tickets for the school play. Student	Marc sold 563 tickets for the school play. Student tickets cost 12 dollars each and none. In total, how many student tickets did he sell at the school?	Marc sold 5 tickets for the school play. He bought 6 tickets at a yard sale over the weekend. How many tickets did he sell in total?	Marc sold 563 tickets for the school play. Student tickets cost 4 dollars and adult tickets cost 6 dollars. Marc's sales totaled 2840 dollars. How many adult tickets and student tickets did Marc sell?
It took Shalani 32 minutes to reach Matara.	It took Shalani 32 minutes to reach Galle. It took Sanduni 2 minutes less than Piyal to reach Galle. How many minutes did it take Piyal and Sanduni to travel to Galle?	It took Sanduni 32 minutes to reach Matara. It took Sanduni 3) minutes less than Piyal to reach the distance they have traveled. How many minutes did it take Piyal and Sanduni to travel to the distance they had?	It took Shalani 32 minutes to reach Matara. It took Lalani 3 minutes less than Shalani to reach Matara. How many minutes did it take for both of them to go to Matara?

C. Additional Experiments

We also experimented with zero-shot MWP generation using GPT-4 (gpt-4-0613) as a baseline, despite it not being a sequence-to-sequence model, given its reported zero-shot generation capabilities. We used the following prompt: "*You are an assistant that creates math word problems based on the starting words provided. Always adhere to the language used in these starting words.*". With respect to the BLEU and METEOR metrics, GPT-4 performed poorly, especially for low-resource languages. A closer analysis of the generated English MWPs revealed instances of hallucination, which explains the low results.

We explored a translation-based approach as well. Initially, we generated English MWPs using the mBART50 baseline model and subsequently employed the "nllb-200-distilled-600 m" model [50] to translate the generated MWPs into the other eight languages. As indicated in Table IX, this method yielded lower-quality results compared to the direct generation of MWPs in the respective languages. The translation approach, despite its potential for broad language coverage, falls short in preserving the intricacies and mathematical precision required for high-quality MWP generation across languages.

D. Human Evaluation

While BLEU-4 and METEOR scores can indicate the quality of generated text in general, they do not have specific capabilities to determine whether the generated MWPs satisfy mathematical constraints. Therefore, we conducted a comprehensive human evaluation of the quality of the generated MWPs.[6] Our primary objective was to identify the impact of the constraint-based text generation model.

The list of errors commonly identified in the generated MWPs is given in Table XI, along with examples of each type. Out of these, the first five errors have been introduced by Niyarepola et al. [12]. They have categorized all errors related to mathematical validity into the *Math constraints* category. In our analysis, this error type is sub-categorized in order to assess the effectiveness of the constraint-based model in preserving the mathematical validity of the generated MWPs.

For the human evaluation, we used a consistent sample of 100 MWPs randomly selected from the Simple dataset. The same corresponding versions of these MWPs were used for each language, ensuring fair comparison across different models and languages. Additionally, we used only the MWPs generated by the mBART50 model, as it had shown the best performance according to automatic metrics. Each set of MWPs was evaluated by three evaluators (who were not part of the manual MWP translation or manual translation evaluation). The evaluators were provided with the set generated by the baseline model, the corresponding set generated by the constraint-based model and a guideline document about the evaluation. The results are reported in Table XII as the average error percentage for each error category.

Human evaluation results generally align with the findings from automatic metrics. We observed reductions in mathematical errors, demonstrating the effectiveness of the constraint-based approach in generating mathematically coherent and logical MWPs. We noted an exception in the Albanian and Sinhala MWPs generated by the constraint-based approach, where the number of mathematical errors increased. The automatic metrics also showed this deviation, with a drop of 2.26 and 0.31 in the BLEU-4 score for Albanian and Sinhala, respectively.

E. Qualitative Evaluation

Table XIII shows the questions generated by the baseline and the constraint-based models for the seed 'Tom got 3 eggs'.

The baseline model produces a question, '*Tom got 3 eggs but 4 were actually damaged. How many good eggs did Tom get?*' leading to an unrealistic answer of *-1*. This suggests a failure to observe a critical constraint: egg count cannot be negative.

Conversely, the constraint-based model, employing the 'Equation Consistency Constraint', formulates a more logical problem: '*Tom got 3 eggs but 2 were actually damaged. How many good eggs did Tom get?*'.

There were instances in which both the baseline and constraint-based models failed to generate high-quality MWPs, as demonstrated in the examples in Table XIV.

Example 1: The baseline model introduces an unrelated entity, a park, demonstrating an issue of hallucination. Furthermore, it generates a problem that lacks sufficient information necessary for a solution. The constraint-based model, while maintaining relevance to the seed, also lacks mathematical validity due to insufficient information for a solution.

[6]This evaluation was not carried out for Assamese, since we could not find human evaluators.

Example 2: The baseline model introduces irrelevant information, demonstrating an issue of hallucination. On the other hand, the constraint-based model alters parts of the input seed, specifically reducing the number of tickets sold, demonstrating an issue of context maintenance. The expected MWP is much longer, which may pose a challenge for both the models. Generating longer sequences while maintaining context and coherence is a known challenge in Natural Language Generation [51], [52].

Example 3: Both models replace the character Shalani with Piyal and introduce a new character, Sanduni, indicating an issue of context maintenance. The constraint-based model also introduces a grammatical error, indicating a language quality issue. Both models fail to generate the expected output, which involves subtraction and addition. This could be due to the models' inability to handle longer sentence structures and maintain context.

In summary, we note that the likelihood of generating erroneous MWPs increases with the problem length. The complexity of the sentence structure and lack of sufficient information in the input seed also contribute to the generation of flawed problems.

VII. Conclusion

This paper presented the multiMWP dataset consisting of nine languages (with six being low-resource) for MWP generation. Using this dataset, we extensively evaluated the utility of prMSLMs for auto-regressive text generation across the nine languages. Our findings reveal that the autoregressive text generation capabilities of prMSLMs are very promising, even for low-resource languages. However, their performance depends on the interplay of language coverage in the model, the existence of similar languages, as well as language script.

We also presented a constraint-based MWP generation model and experimented with multilingual MWP generation. The model that combines both these extensions is our pick for MWP generation.

Even though this model improves the quality of the MWPs, we still see some issues in the generated MWPs, especially related to mathematical constraints. In future work, we plan to explore the use of decoder-based LMs such as Llama [53] and DeepSeek [54], which are commonly known as Large Language Models (LLMs). In particular, there have been efforts to improve the mathematical reasoning abilities of LLMs [55], [56]. We plan to experiment with such techniques in the future. As mentioned earlier, ChatGPT, which is a prominent commercial LLM, performed poorly on low-resource languages. We believe that, by improving the multilingual capabilities of LLMs as done by previous research for tasks such as Machine Translation [48], [57], this limitation can be mitigated to a certain extent. Exploring these approaches—potentially in combination with LLM agents in a pipeline system—could mitigate existing issues and further enhance MWP generation. Finally, we encourage researchers to extend our dataset to additional languages, fostering further advancements in multilingual MWP research.

Appendix A
Ethical Considerations

Workers were compensated either by paying university stipulated rates or were made co-authors of the paper. We obtained permission to use Niyarepola et al.'s [12]'s dataset. Wang et al. [19] and Tennage et al. [49] privately provided their code upon request. The dataset contains only elementary-level Mathematics questions. A generative model trained with this data cannot cause any additional harm than what can be already caused by the prMSLMs we used.

Appendix B
Supplementary Materials

Table XV shows some of the hyper-parameters used in training the models.

TABLE XV
Model Hyper-Parameters

Approach	Optimizer	lr	Batch size
Baseline	Adam	0.0001	8
Constraint-based	Adam	[1e-5 ,1e-4]	8

Acknowledgment

The authors would like to thank Dr. Rishemjit Kaur and Dr. Mayuri Chabukdhara for their support in finding Assamese speakers, as well as Thillainathan Sarubi for her support for Tamil MWP data and Dr. Ravi Shekhar for his support for Hindi data. We also thank Accelr Logic, Sri Lanka and 'AWS cloud credit for research' program (grant received by Surangika Ranathunga) for providing GPU resources.

References

[1] S. Wijesundera and S. Yatigammana, "Mathematics teachers' beliefs on students' low achievements at junior secondary level of education in Sri Lanka and their implications for school based teacher development (SBTD)," 2021, doi: 10.2139/ssrn.3809030.

[2] B. R. Acharya, "Factors affecting difficulties in learning mathematics by mathematics learners," *Int. J. Elementary Educ.*, vol. 6, no. 2, pp. 8–15, 2017.

[3] L. J. Rylands and C. Coady, "Performance of students with weak mathematics in first-year mathematics and science," *Int. J. Math. Educ. Sci. Technol.*, vol. 40, no. 6, pp. 741–753, 2009.

[4] S. Sethi, "A study on problems in teaching mathematics at upper-primary level of khurda district, odisha," *DEI-FOERAA, Dayalbagh Educ. Instit.-Faculty Educ. Res. Abstr. Art.: Res. J. Educ.*, vol. 1, pp. 1–7, 2021, Dayalbagh, Agra, India.

[5] S. Maghnouj, E. Fordham, C. Guthrie, K. Henderson, and D. Trujillo, *OECD Reviews of Evaluation and Assessment in Education: Albania*.OECD Publishing, Paris, France, 2020, doi: 10.1787/d267dc93-en.

[6] Q. Zhou and D. Huang, "Towards generating math word problems from equations and topics," in *Proc. 12th Int. Conf. Natural Lang. Gener.*, 2019, pp. 494–503.

[7] G. Kurdi, J. Leo, B. Parsia, U. Sattler, and S. Al-Emari, "A systematic review of automatic question generation for educational purposes," *Int. J. Artif. Intell. Educ.*, vol. 30, pp. 121–204, 2020.

[8] S. Ranathunga and N. de Silva, "Some languages are more equal than others: Probing deeper into the linguistic disparity in the NLP world," in *Proc. 2nd Conf. Asia-Pacific Chapter Assoc. Comput. Linguistics 12th Int. Joint Conf. Natural Lang. Process.*, 2022, pp. 823–848.

[9] S. Ranathunga et al., "Neural machine translation for low-resource languages: A survey," *ACM Comput. Surv.*, vol. 55, no. 11, pp. 1–37, 2023.

[10] A. Paullada, I. D. Raji, E. M. Bender, E. Denton, and A. Hanna, "Data and its (DIS) contents: A survey of dataset development and use in machine learning research," *Patterns*, vol. 2, no. 11, 2021, Art. no. 100336.

[11] S. Ranathunga, N. De Silva, D. Jayakody, and A. Fernando, "Shoulders of giants: A look at the degree and utility of openness in NLP research," in *Proc. 62nd Annu. Meeting Assoc. Comput. Linguistics (Volume 2: Short Papers)*, 2024, pp. 519–529.

[12] K. Niyarepola, D. Athapaththu, S. Ekanayake, and S. Ranathunga, "Math word problem generation with multilingual language models," in *Proc. 15th Int. Conf. Natural Lang. Gener.*, 2022, pp. 144–155.

[13] P. Joshi, S. Santy, A. Budhiraja, K. Bali, and M. Choudhury, "The state and fate of linguistic diversity and inclusion in the NLP world," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 6282–6293.

[14] Y. Liu et al., "Multilingual denoising pre-training for neural machine translation," *Trans. Assoc. Comput. Linguistics*, vol. 8, pp. 726–742, 2020.

[15] L. Xue et al., "mT5: A massively multilingual pre-trained text-to-text transformer," in *Proc. 2021 Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2021, pp. 483–498.

[16] A. Fan et al., "Beyond english-centric multilingual machine translation," *J. Mach. Learn. Res.*, vol. 22, no. 107, pp. 1–48, 2021.

[17] R. Dabre, H. Shrotriya, A. Kunchukuttan, R. Puduppully, M. M. Khapra, and P. Kumar, "IndicBART: A pre-trained model for indic natural language generation," in *Proc. Findings Assoc. Comput. Linguistics: ACL 2022*, 2022, pp. 1849–1863.

[18] V. Liyanage and S. Ranathunga, "Multi-lingual mathematical word problem generation using long short term memory networks with enhanced input features," in *Proc. 12th Lang. Resour. Eval. Conf.*, 2020, pp. 4709–4716.

[19] Z. Wang, A. Lan, and R. Baraniuk, "Math word problem generation with mathematical consistency and problem context constraints," in *Proc. 2021 Conf. Empirical Methods Natural Lang. Process.*, 2021, pp. 5986–5999.

[20] W. Nekoto et al., "Participatory research for low-resourced machine translation: A case study in African languages," in *Proc. Findings Empirical Methods Natural Lang. Process.*, pp. 2144–2160, 2020.

[21] R. Koncel-Kedziorski, S. Roy, A. Amini, N. Kushman, and H. Hajishirzi, "MAWPS: A math word problem repository," in *Proc. 2016 Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2016, pp. 1152–1157.

[22] M. J. Hosseini, H. Hajishirzi, O. Etzioni, and N. Kushman, "Learning to solve arithmetic word problems with verb categorization," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2014, pp. 523–533.

[23] R. Koncel-Kedziorski, H. Hajishirzi, A. Sabharwal, O. Etzioni, and S. D. Ang, "Parsing algebraic word problems into equations," *Trans. Assoc. Comput. Linguistics*, vol. 3, pp. 585–597, 2015.

[24] S. Roy and D. Roth, "Reasoning about quantities in natural language," *Trans. Assoc. Comput. Linguistics*, vol. 3, pp. 1–13, 2015.

[25] S. Roy and D. Roth, "Solving general arithmetic word problems," in *Proc. 2015 Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1743–1752.

[26] D. Huang, S. Shi, C.-Y. Lin, J. Yin, and W.-Y. Ma, "How well do computers solve math word problems? large-scale dataset construction and evaluation," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics (Volume 1: Long Papers)*, 2016, pp. 887–896.

[27] Y. Wang, X. Liu, and S. Shi, "Deep neural solver for math word problems," in *Proc. 2017 Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 845–854.

[28] N. Kushman, Y. Artzi, L. Zettlemoyer, and R. Barzilay, "Learning to automatically solve algebra word problems," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics (Volume 1: Long Papers)*, 2014, pp. 271–281.

[29] S. Upadhyay and M. Chang, "Annotating derivations: A new evaluation strategy and dataset for algebra word problems," in *Proc. 15th Conf. Eur. Chapter Assoc. Comput. Linguistics: Vol. 1, Long Papers*, 2017, pp. 494–504.

[30] S. Shi, Y. Wang, C. Lin, X. Liu, and Y. Rui, "Automatically solving number word problems by semantic parsing and reasoning," in *Proc. 2015 Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1132–1142.

[31] A. Amini, S. Gabriel, S. Lin, R. Koncel-Kedziorski, Y. Choi, and H. Hajishirzi, "MathQA: Towards interpretable math word problem solving with operation-based formalisms," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2019, pp. 2357–2367.

[32] R. Koncel-Kedziorski, I. Konstas, L. Zettlemoyer, and H. Hajishirzi, "A theme-rewriting approach for generating algebra word problems," in *Proc. 2016 Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 1617–1628.

[33] K. Nandhini and S. R. Balasundaram, "Math word question generation for training the students with learning difficulties," in *Proc. Int. Conf. Workshop Emerg. Trends Technol.*, 2011, pp. 206–211.

[34] O. Polozov, E. O'Rourke, A. M. Smith, L. Zettlemoyer, S. Gulwani, and Z. Popović, "Personalized mathematical word problem generation," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, 2015, pp. 381–388.

[35] K. Wang and Z. Su, "Dimensionally guided synthesis of mathematical word problems," in *Proc. Int. Joint Conf. Artif. Intell.*, 2016, pp. 2661–2668.

[36] A. A. Bekele, "Automatic generation of amharic math word problem and equation," *J. Comput. Commun.*, vol. 8, no. 8, pp. 59–77, 2020.

[37] V. Liyanage and S. Ranathunga, "A multi-language platform for generating algebraic mathematical word problems," in *Proc. 14th Conf. Ind. Inf. Syst.*, IEEE, 2019, pp. 332–337.

[38] D. Liu et al., "GLGE: A new general language generation evaluation benchmark," in *Proc. Findings Assoc. Comput. Linguistics: ACL-IJCNLP 2021*, 2021, pp. 408–420.

[39] T. Cao, S. Zeng, S. Zhao, M. Mansur, and B. Chang, "Generating math word problems from equations with topic consistency maintaining and commonsense enforcement," in *Proc. Int. Conf. Artif. Neural Netw.*, Springer, 2021, pp. 66–79.

[40] Z. Zhou et al., "Learning by analogy: Diverse questions generation in math word problem," in *Proc. Findings Assoc. Comput. Linguistics: ACL 2023*, 2023, pp. 11091–11104.

[41] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang, "Language-agnostic BERT sentence embedding," in *Proc. 60th Annu. Meeting Assoc. Comput. Linguistics (Vol. 1: Long Papers)*, 2022, pp. 878–891.

[42] A. Conneau et al., "Unsupervised cross-lingual representation learning at scale," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 8440–8451.

[43] O. Bojar et al., "Findings of the 2016 conference on machine translation," in *Proc. 1st Conf. Mach. Transl.: Volume 2, Shared Task Papers*, Berlin, Germany, 2016, pp. 131–198. [Online]. Available: https://aclanthology.org/W16-2301/

[44] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, 2002, pp. 311–318.

[45] A. Lavie and A. Agarwal, "METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments," in *Proc. Second Workshop Stat. Mach. Transl.*, 2007, pp. 228–231.

[46] E. Lee et al., "Pre-trained multilingual sequence-to-sequence models: A hope for low-resource language translation?," in *Proc. Findings Assoc. Comput. Linguistics: ACL 2022*, 2022, pp. 58–67. [Online]. Available: https://aclanthology.org/2022.findings-acl.6

[47] J. Hu, S. Ruder, A. Siddhant, G. Neubig, O. Firat, and M. Johnson, "XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 4411–4421.

[48] F. Yuan, S. Yuan, Z. Wu, and L. Li, "How vocabulary sharing facilitates multilingualism in LLaMA?," *Findings Assoc. Comput. Linguistics*, 2024, pp. 12 111–12 130.

[49] P. Tennage, A. Herath, M. Thilakarathne, P. Sandaruwan, and S. Ranathunga, "Transliteration and byte pair encoding to improve Tamil to Sinhala neural machine translation," in *Proc. 2018 Moratuwa Eng. Res. Conf.*, IEEE, 2018, pp. 390–395.

[50] N. Team et al., "No language left behind: Scaling human-centered machine translation," 2022, *arXiv:2207.04672*.

[51] J. Guan, X. Mao, C. Fan, Z. Liu, W. Ding, and M. Huang, "Long text generation by modeling sentence-level and discourse-level coherence," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics 11th Int. Joint Conf. Natural Lang. Process. (Volume 1: Long Papers)*, 2021, pp. 6379–6393.

[52] N. Malkin, Z. Wang, and N. Jojic, "Coherence boosting: When your pretrained language model is not paying enough attention," in *Proc. 60th Annu. Meeting Assoc. Comput. Linguistics (Volume 1: Long Papers)*, Dublin, Ireland, 2022, pp. 8214–8236.

[53] H. Touvron et al., "LLaMA: Open and efficient foundation language models," 2023, *arXiv:2302.13971*.

[54] DeepSeek-AI et al., "DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning," 2025, *arXiv:2501.12948*.

[55] R. Yamauchi, S. Sonoda, A. Sannai, and W. Kumagai, "LPML: LLM-prompting markup language for mathematical reasoning," 2023, *arXiv:2309.13078*.

[56] Q. Zhong, K. Wang, Z. Xu, J. Liu, L. Ding, and B. Du, "Achieving >97% on GSM8K: Deeply understanding the problems makes LLMs perfect reasoners," 2024, *arXiv:2404.14963*.

[57] K. Peng et al., "Towards making the most of ChatGPT for machine translation," in *Proc. Findings Assoc. Comput. Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore, Association for Computational Linguistics, Dec. 2023, pp. 5622–5633. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.373/